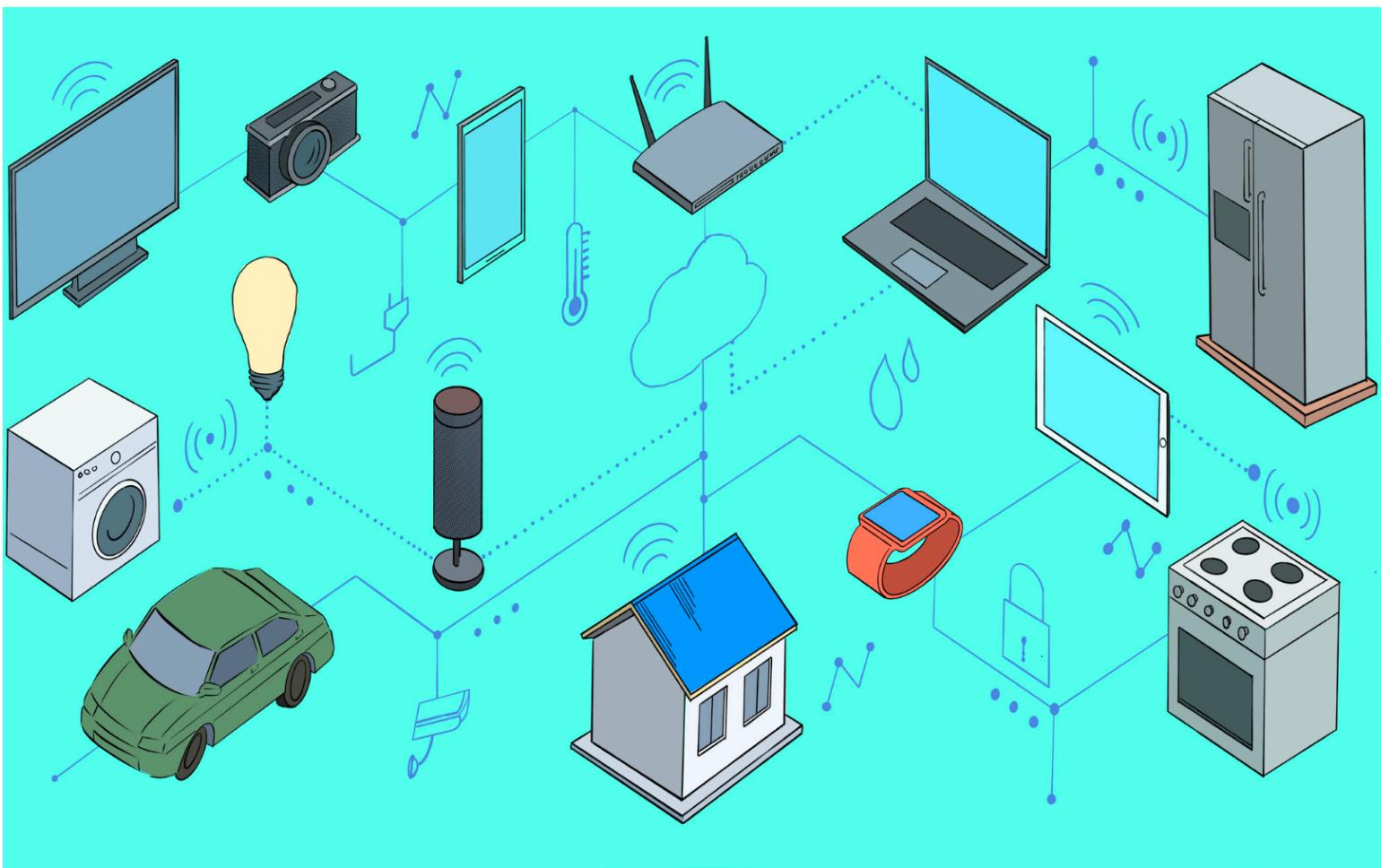


# Computer Science

for Advanced Secondary Schools

Student's Book **Form Six**



Tanzania Institute of Education

Property of the Government of the United Republic of Tanzania, Not for Sale

FOR ONLINE USE ONLY  
DO NOT DUPLICATE

# Computer

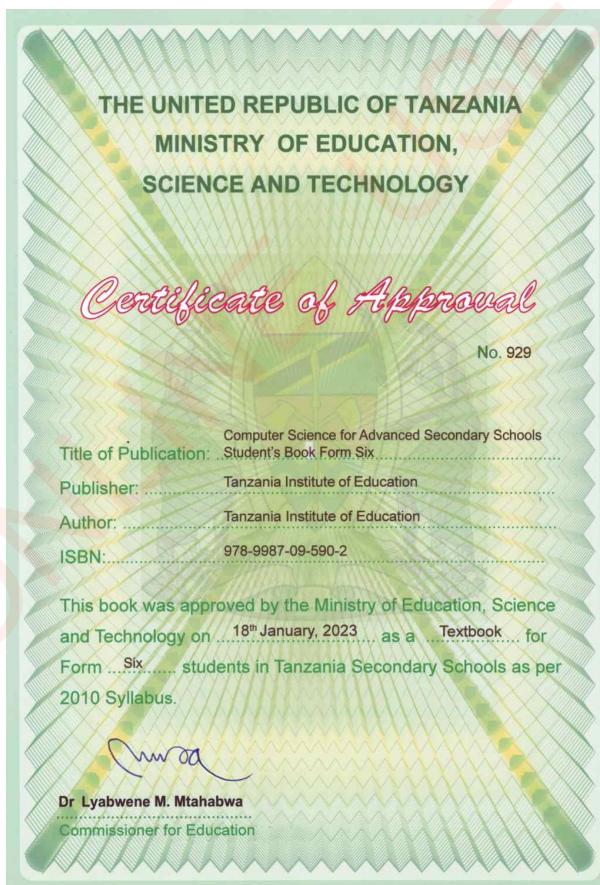
# Science

for

## Advanced Secondary Schools

### Student's Book

### Form Six



Tanzania Institute of Education

FOR ONLINE USE ONLY

© Tanzania Institute of Education, 2022 DUPLICATE

Published 2022

ISBN: 978-9987-09-590-2

Tanzania Institute of Education  
P.O. Box 35094  
Dar es Salaam, Tanzania

Mobile numbers: +255 735 041 170 /+255 735 041 168  
E-mail: director.general@tie.go.tz  
Website: www.tie.go.tz

All rights reserved. No part of this textbook may be reproduced, stored in any retrieval system, or transmitted in any form or by any means, whether electronic, mechanical, photocopying, recording, or otherwise, without prior written permission of the Tanzania Institute of Education.

<b>Acronyms and Abbreviations.....</b>	<b>v</b>
<b>Acknowledgements .....</b>	<b>vii</b>
<b>Preface.....</b>	<b>viii</b>
 <b>Chapter One</b>	
<b>Data structures and algorithms .....</b>	<b>1</b>
Concept of data structure .....	1
Primitive data structure .....	3
Non-primitive data structure .....	4
Static data structure.....	6
Pointers .....	11
Dynamic data structure .....	23
Algorithms .....	34
Recursive algorithms .....	37
Iterative algorithm.....	40
Searching algorithm .....	42
Sorting algorithm .....	45
Chapter summary .....	55
Revision exercise .....	56
 <b>Chapter Two</b>	
<b>Information systems.....</b>	<b>59</b>
Overview of Information Systems....	59
Types of information systems .....	70
Database systems .....	74
Types of database models .....	77
Relational database .....	79
Entity-relationship model.....	90
Normalisation.....	96
Database management .....	103
General database administration.....	110
Chapter summary .....	111
Revision exercise .....	111
 <b>Chapter Three</b>	
<b>Data communication and networking .....</b>	<b>113</b>
Concept of data communication .....	113
Computer networks .....	127
Client-server network architecture .	141
Transmission media .....	168
Chapter Summary .....	193
Revision exercise .....	198
 <b>Chapter Four</b>	
<b>Visual programming .....</b>	<b>200</b>
Concept of visual programming.....	200
Visual Basic .NET and the .NET Framework .....	201
Components of the Visual Studio 2022 IDE.....	209
Programming in Visual Basic .NET .....	226
Visual basic control structures .....	236
Visual Basic .NET procedures and functions.....	248
Linking visual Basic .NET form to data source .....	252
Chapter summary .....	263
Revision exercise .....	264
 <b>Chapter Five</b>	
<b>Computer security and privacy .....</b>	<b>265</b>
The concept of computer security ..	265
Data protection.....	268
Attacks .....	275
Authorization and access control techniques .....	290

Data loss prevention .....	310
Vulnerability analysis and penetration testing.....	313
Security in cloud computing and the Internet of things (IoT) .....	320
Physical security .....	323
Software piracy and copyright.....	326
Ethical and legal issues .....	330
Chapter summary .....	332
Revision exercise .....	333
 <b>Chapter Six</b>	
<b>Information technology career and environment .....</b>	<b>334</b>
IT career opportunities .....	334
IT and environment.....	341
Chapter summary .....	345
Revision exercise .....	346
 <b>DChapter Seven</b>	
<b>Emerging technologies and trends .....</b>	<b>347</b>
Concept of emerging technologies and trends .....	347
Industrial revolutions .....	350
Emerging technologies and their applications .....	355
Impact of emerging technologies....	374
Chapter summary .....	377
Revision exercise .....	378
 <b>Glossary .....</b> 380	
<b>Bibliography .....</b>	<b>384</b>
<b>Index .....</b>	<b>386</b>

## Acronyms and Abbreviations

1NF	First Normal Form
2NF	Second Normal Form
3D	Three (3) Dimension
3NF	Third Normal Form
4D	Four (4) Dimension
4IR	Fourth Industrial Revolution
5D	Five (5) Dimension
ACK	Acknowledgment
ACL	Access Control List
ADSL	Asymmetrical Digital Subscriber Line
AI	Artificial Intelligence
AR	Augmented Reality
ASCII	American Standard Code for Information Interchange
CAD	Computer-Aided Design
CIDR	Classless Inter-Domain Routing
CPS	Cyber-Physical Systems
CRC	Cyclic Redundancy check
DBMS	Database Management System
DHCP	Dynamic Host control protocol
DoS	Denial of service
DSL	Digital Subscriber Line
DSS	Decision Support System
EIS	Executive Information System
ERD	Entity Relational Diagram
FDM	Frequency Division Multiplexer
FTP	File Transfer Protocol
IoT	Internet of Things

**FOR ONLINE USE ONLY  
DO NOT DUPLICATE**

IS	Information System
ISO	International Standard for Organisation
LAN	Local Area Network
MAC	Media Access Control
MAN	Metropolitan Area Network
MAU	Media Access Unit, also Multistation Access Unit,
ML	Machine Learning
NAT	Network Address Translation
NIC	Network interface card
OSI	Open system interconnect
PAN	Personal Area Network
PAT	Port Address Translation
PC	Personal computer
SDLC	System Development Life Cycle
SMS	Short Message Service
SQL	Structured Query Language
SYN	Synchronise
SYN-ACK	Synchronise acknowledgment
TDM	Time Division Multiplexer
TPS	Transaction Processing System
UDP	User datagram protocol
VPN	Visual private network
VR	Virtual Reality
WAN	Wide Area Network
WLAN	Wireless Area Network

The Tanzania Institute of Education (TIE) would like to acknowledge the contributions of all organisations and individuals who participated in designing and developing this textbook. In particular, TIE wishes to thank the University of Dar es Salaam (UDSM), Mzumbe University (MU), The Open University of Tanzania (OUT), Dar es Salaam Institute of Technology (DIT), The Mwalimu Nyerere Memorial Academy (MNMA), the National Examinations Council of Tanzania (NECTA), School Quality Assurance (SQA) Department, Teachers' Colleges and Secondary Schools. Besides, the following individuals are acknowledged:

**Writers:** Mr Chacha M. Nsaho (Mpwapwa Teachers' College),  
 Dr Edephonce N. Nfuka (OUT), Dr-Eng. Morice D. Tegeje (MU),  
 Dr Fabian G. Mahundu (MNMA), Mr Justine J. Nnko (DIT),  
 Mr Leonard T. Mkama (Kilakala S.S), & Mr Robert Chassama  
 (Morogoro Teachers' College).

**Editors:** Dr Jabhera Matogoro and Dr Ally S. Nyamawe, (UDOM),  
 Dr Theophil B. Assey, (IAA), Ms Zawadi L. Tito, (Iyunga Technical  
 Secondary School), Mr Mussa M. Gunda (Korogwe TC),  
 Dr Michael A. Mashauri (UDSM-Language) & Dr Rafiki Y. Sebonde  
 (UDOM-Language).

**Designer:** Ms Eva G. Kwai

**Illustrators:** Chacha M. Nsaho & Alama Art and Media Production Co. Ltd

**Coordinator:** Chacha M. Nsaho (Mpwapwa Teachers' College)

TIE also appreciates secondary school teachers and students who participated in the trial phase of the manuscript of this book. Likewise, TIE would like to thank the Ministry of Education, Science and Technology for facilitating the writing and printing of this textbook.



Dr Aneth A. Komba

**Director General**

**Tanzania Institute of Education**

FOR ONLINE USE ONLY

 No Replicate

This textbook, Computer Science for Advanced Secondary Schools Student's Book Form Six, is explicitly written for form six students in the United Republic of Tanzania.

It is written following the 2010 Computer Science Syllabus for Advanced Secondary Education, Form V-VI issued by the then Ministry of Education and Vocational Training. The book consists of seven chapters: Data structure and algorithms, Information systems, Data communications and networking, Visual programming, Computer security and privacy, Information technology career and environments, and Emerging technologies. In addition to textual descriptions, each chapter contains illustrations, activities, and exercises. You are encouraged to do all activities and exercises in this book and other assignments provided by your teacher. This will enable you to develop the intended competencies.

Tanzania Institute of Education

# Data structures and algorithms

## Introduction

In form five, you learnt about problem-solving, the concepts of algorithms and pseudocode, and programming. You demonstrated how to program in C++ language, which is essential in understanding data structures and algorithms. Such data structures and algorithms are essential in developing efficient software programs. In this topic, you will learn how data may be structured and instructions sequenced in algorithms, programs and the relationship between appropriate data and control structures. Studying data structures and algorithms helps to understand the basic concepts involved in organising and storing data and the relationship among the data sets. It helps to determine how information is stored, retrieved, and modified in a computer's memory. The competencies developed from data structures and algorithms will help you to write efficient programs intended to retrieve and store massive data of various types.

## Concept of data structure

The data structure is a branch of Computer Science that deals with the organisation, flow, and data storage which builds the logical relationships between data elements to maximise the efficiency of the computer programs or processes. A well-designed data structure allows various critical operations to be performed using as few resources, both execution time and memory space, as possible.

The way of organising data affects the performance of a program for different tasks. Computer programmers decide which data structures to use based on the nature of the data and the processes to be performed.

The data structure organises data items based on the relationship between the data elements in the computer program. For example, attributes such as the model, name, car number, engine capacity and number, transmission type, colour can identify a car. These structured variables depend on each other to identify the exact car. Similarly, the data structure is an organised set of interrelated variables that form a system's elementary constituent.

Generally, the data structure links the programmer's viewpoint of data and the implementation of these data in computers. It comprises the storage methods and one or more algorithms used to access or modify data.

## Importance of data structure

Nowadays, data structure is essential in programming for producing efficient programs. The following are the importance of data structure:

- It makes it easy for users to access and work with the data they need in the appropriate ways.
- It frames information organisation so that machines and humans can better understand it.
- It offers the means to manage well a considerable quantity of data.
- It provides a diverse level of data organisation.
- It provides operations on a data group such as, adding an item and looking for the highest precedence item.
- It tells how data can be stored and accessed at its fundamental level.
- It enables fast searching and sorting of data.

## Goals of data structure

During the data structure implementation, its operations have two goals. These goals are:

- Correctness:** For all inputs within a program, the data structures are designed to operate correctly for a specific problem intended to be solved.
- Efficiency:** Data are processed at the required speed without using much computer resources, and stored in a specific memory location.

## Features of data structures

Some of the essential features of data structures are described as follows:

**Reusability:** The programmer requires many resources to develop any software, making it an expensive enterprise. However, if the software designed is reusable and adaptable, it can be implemented in most future applications. Thus, by implementing quality data structures, it is possible to develop reusable software which tends to be cost-effective and time-saving.

**Robustness:** Generally, all computer programmers expect to produce software that generates correct output for every possible input provided to it and executes efficiently on all hardware platforms. This kind of robust software must manage both valid and invalid inputs.

**Adaptability:** Developing software projects such as web browsers, word processors, and search engines involve bulky software systems that have worked or been accomplished appropriately and efficiently for several years. Additionally, software evolves due to continually changing market environments or emerging technologies.

## Factors to consider when selecting a data structure

It is essential to choose the appropriate data structure for each task. The factors to consider when selecting the proper data structure include the following:

- Type of information to be stored
- The uses of data or information
- Ways or methods to organise the data
- The aspects of memory and storage arrangement plan and their management

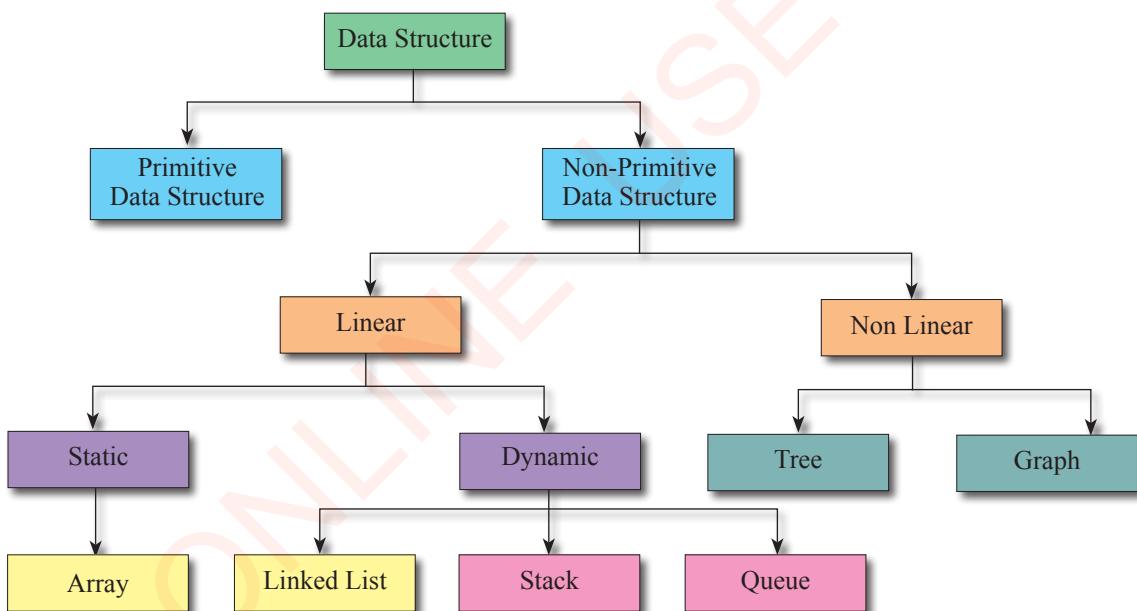
- (e) Storage devices for data after being generated

the structure; homogeneous and non-homogeneous (heterogeneous) based on the similarity of structures.

### **Classification of data structure**

The data structure can broadly be categorised into two distinct types: A primitive data structure and a non-primitive data structure. In other books, data structures have been classified based on various aspects, for instance, linear data structure and non-linear data structure based on the arrangement or organisation of the structure; static data structure and dynamic data structure based on the memory size of

In this book, as depicted in Figure 1.1, the data structure is broadly classified based on the nature of the operation that can be conducted on the unit of respective data. The primitive data structure can store only values of the same data type, while non-primitive data structures can store values with more than one data type. Figure 1.1 shows the classification of primitive and non-primitive data structures with examples.



**Figure 1. 1: Types of data structure**

### **Primitive data structure**

The primitive data structure is the fundamental category of data type. Its operations and functionalities are only performed at the machine level because they consist of numbers and characters built into a program. Any operation can be carried out in these types of data items. They are also called pre-defined or built-in data structures.

They are named in that way because the user can not change or make any alteration as it is handled at the machine level. They are created without the support of other structures as tools. They are directly used in the program for any specific task. The user does not define them. Examples of primitive data structures are integers, floats, doubles, characters, Boolean, pointers, and strings. Primitive data structure stores only one type of data or may contain empty values. Figure 1.2. illustrates the sub-category of primitive data structures.

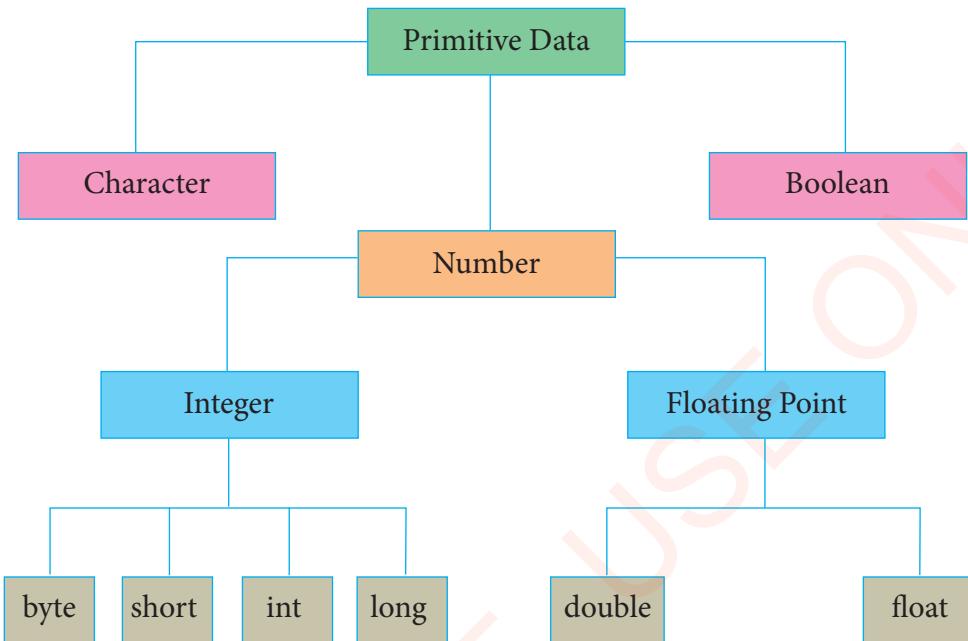


Figure 1. 2: Classification of primitive data structure

### Number:

**Integer:** Represents numbers without decimal points, for example, 5, 10, 200, 1254. It has a 32 bits size.

**Float:** Represent variables defined with fractional values. These variables have decimal points, for example, 10.2, 22.7, 1076.3. It has a 32 bits size containing up to 7 decimals.

**Double:** It is the same as the float type but has a 64 bits size and contains up to 15 decimals. For example, -1.79769313486231E308 to -4.94065645841247E324 for negative values and 4.94065645841247E-324

to 1.79769313486231E308 for positive values.

**Character:** represents the variables that use symbols/ characters or letters, for example, r, R, h, A, y, X, “-”. It has a 16 bits size.

**Boolean:** Represent logical values such as; true or false, yes or no, 1 or 0. It has a 16 bits size.

### Non-primitive data structure

The non-primitive data structure is a user-defined data structure that can hold several values in adjoining or

## FOR ONLINE USE ONLY

random locations. They must contain some values, i.e. they cannot be empty. It should have a specific function. The non-primitive data structure is further classified into two categories, namely; linear and non-linear data structures based on the arrangement and organisation of data.

### **Linear data structure**

The linear data structure stores data in a sequence, one after another and the data is accessed from one place and continues to others sequentially. The linear data structure is the data structure in which its elements are linked with the subsequent one sequentially. A data structure is linear if its features are combined to form any specific order. There are two techniques for representing linear structure within a memory. The first technique is to represent the linear relationships among all the elements described using linear memory location. These linear structures are termed as arrays. The second technique represents the linear relationship among all the elements using pointers or links. This technique of representing linear structures is termed linked lists. Furthermore, the linear data structure is classified into two types based on how data can be stored: static and dynamic data structures. The arrays, stack, linked list, and queue are examples of linear data structures.

### **Static data structure**

The static data structure is a method of storing data where the amount of data stored and the memory used to hold it are fixed. In a static data structure, the content of the data structure can be modified without changing the

memory space allocated to it. The size of the structure is fixed and permanent at compile time. Accessing individual data elements within a static structure is speedy as their memory location is fixed. A static data structure stores a set of amounts of data, usually user-defined. An example of a static data structure is an array.

### **Dynamic data structure**

In a dynamic data structure, the size of the structure is not fixed and can be modified when operations are executed on it. Dynamic data structures aim to facilitate the change of data structures in the run time. Examples of dynamic data structures are stacks, Linked lists, and queues.

### **Non-linear data structure**

In a non-linear data structure the element may link to more than one element, although their data items are not sequential. This structure mainly represents data with a hierarchical relationship among various components. An example of a non-linear data structure is a tree.

### **Exercise 1.1**

#### **Answer the following questions:**

1. What is the data structure?
2. Explain the importance of data structure.
3. Explain any three features of data structures.
4. Briefly describe the classification of data structure.

5. Kazolamimba Company wants to decide on a data structure to implement the program. Explain how you would help them to choose the best data structure.
6. Describe the types of data structures.
7. With examples, describe primitive data structures.
8. Give examples of non-primitive data structures.

## Static data structure



### Think about the following:

Arrangements of crates of soda in the shops or depots; arrangements of soda in the crate, and the addition of some bottles of soda than crates' capacity of the container. Also, think about other items packed in cartons, dozen, and trays. Can you add more items than its capacity?

Observe their arrangement and tell if you can add more items than the size of their containers. Give reasons to support your answer.

### Meaning of static data structure

The static data structure has a fixed size of the memory structure. The content of the data structure can be modified without changing the assigned memory space.

### Advantages and disadvantages of static data structure

Static data structure offers benefits and limitations that a programmer should consider when deciding to use them.

### Advantages of static data structures

The following are the advantages of static data structure:

**Easier to comprehend:** It is easier to define and use a static data structure than a dynamic one.

**Size:** It has a fixed size.

**Faster:** The memory for static data is assigned during the compilation time, making the program execute more quickly than when it has to stop and allocate the required memory each time. An array index can randomly access any element of an array.

### Disadvantages of static data structure

**Memory wastage:** Declaring an array with a large size is a memory wastage. For example, if you declare an array of 10 and only a tiny portion less than the declared is used, the rest of the memory will be wasted. Therefore, it can waste a lot of space if the estimation is too high.

**Insufficient memory space:** Declaring an array with a fixed size leads to a situation where the size of the array is less than what you intended to use. In other words, there is insufficient memory for the program to run with all data required. In another situation, requesting a consecutive large block may not be granted, for example, int y[6]; it has 6 elements, but the program needs 10 memory.

**Slow:** Some algorithms are slower when done statically than when done dynamically. How would you merge the two lists? The solution with static arrays is to define a more extensive array and copy the two arrays into a more considerable array.

**Not ordinary:** More suitable algorithms can be implemented dynamically rather than statically.

### Types of static data structure

As seen earlier, the static data structure is a method of storing data where the amount of data stored and the memory used to hold it are fixed. An example of a static data structure is an array.

### Arrays

The term array is generally used in computer programming to mean a contiguous chunk of memory locations where each memory location holds one fixed-length data item. By this meaning, an array is a physical data structure. However, the array can also mean a logical data type composed of a typically homogeneous collection of data items, each identified by an index number.

An array is a list or table of data with a variable name identifying such a list or table. Each item in the table is called an element. An array can have many dimensions. However, most arrays are either one-dimensional, in which they form a list or can be two-dimensional, visualised as a table. The array is a container that can hold a fixed number of items. These items should be of the same data type. For example, integers only or float only or string only. Most of the data structures make use of arrays to implement their algorithms.

In some programming languages, the size of an array must be established once and for all during the program designing time and cannot change during the execution. Such arrays are called *static arrays*. The chunk of memory that is big enough and sufficient to hold all the values in the array is allocated when the

array is created. After that, elements are accessed using the fixed base location of the array. Static arrays are the fundamental array type in most older procedural languages, such as Fortran, Basic, and C, and many newer object-oriented languages as well, such as Java.

The following are essential terms to understand the concept of an array:

- (a) Element – refers to every item stored in an array.
- (b) Index – used to identify the location of the element.

### Classification of arrays in algorithm and data structure

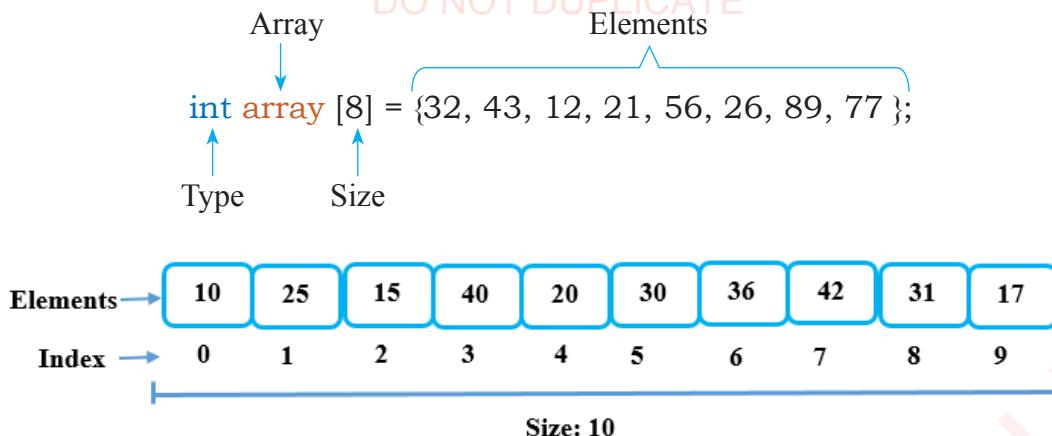
Arrays must be classified before being used in a process. They are classified into two dimensions as follows:

- (a) **One-dimensional array:** This array has one row of elements and is stored in ascending storage location in its table.
- (b) **Multidimensional arrays:** This kind of array includes as many indices as required because they are not bound to two indices or two dimensions. Examples of multidimensional arrays are:
  - (i) Two-dimensional (2-D) arrays or matrix arrays
  - (ii) Three-dimensional (3-D) arrays

### One-dimensional array structure

One-dimensional arrays can be declared in various ways in different languages. For example, from the C++ array declaration, the structure of a one-dimensional array is as follows:

FOR ONLINE USE ONLY  
DO NOT DUPLICATE



As per the previous example, the following are important points to be considered:

- The index starts with 0.
- The array length is 10, which means it can store maximum 10 elements.
- Each element can be accessed via its index. For example, you can fetch an element at index 6 as 36.

When you declare an array in a program, assign initial values to each of its elements by enclosing the values in braces {}.

`int Num[10] = {2, 9, 12, 78, 34, 23, 1, 29, 50, 13};`, this declaration will create the following array:

Num	2	9	12	78	34	23	1	29	50	13
-----	---	---	----	----	----	----	---	----	----	----

**Example 1:** Add all elements of an array using C++ language.

```
#include <iostream>
using namespace std;
int schools [3] = {5, 9, 10};
int n, result=0;
int main ()
{
    for ( n=0 ; n<3 ; n++ )
    {
        result += schools[n];
    }
    cout<< result;
    return 0;
}
```

## FOR ONLINE USE ONLY

## © 2023 TIE

The output for the program is 24 as shown in the snapshot in Figure 1.3.

```
24
Process exited after 0.125 seconds with return value 0
```

**Figure 1.3:** Out of the program to add element of the array

### Two-dimensional array (2-D array)

Two-dimensional (2-D) arrays are indexed by two subscripts, one for the row and one for the column. The Syntax of the two-dimensional array is as follows:

(Data type) (Name of Array)  
[Number of rows] [number of columns];

#### For example:

```
int matrix [7] [7];
```

When you type this statement, the compiler will generate a 2-D array of a matrix which consists of 7 rows and 7 columns.

#### Accessing location of two-dimensional array elements

To store values in a C++ two-dimensional array, the programmer has to specify the number of rows and columns of a matrix. To access each matrix location to store the values, the user must also provide the exact number of rows and columns.

#### For example:

```
int matrix [3][3];
test[0][0] = 45
test[0][1] = 12
test[0][2] = 67
test[1][0] = 78
test[1][1] = 34
test[1][2] = 78
test[2][0] = 89
test[2][1] = 23
test[2][2] = 67
```

#### Entering data in two-dimensional arrays:

A nested loop is used to enter data in 2-D arrays. It depends upon the programmer which loop they want to use, which can be a *While loop* or a *For loop*. The outer loop acts as the number of rows of a matrix and the inner loop acts as the number of columns of a matrix.

Example of a C++ program to enter and display a two-dimensional array is as follows:

```
#include <iostream>
using namespace std;
int main () {
    int test [3][3];
    // For taking integer inputs in a matrix //
    for (int i=0 ; i<3 ; i++)
    {
        for (int j=0 ; j<3 ; j++)
        {
            cout<<"Enter Integer :";
            cin>>test [i][j];
        }
    }
    cout<<endl;
    // use of nested for loop
    // access rows of the array
    for (int i = 0; i < 3;
++i) {
        // access columns of the array
        for (int j = 0; j < 3; ++j) {
            cout << "test[" << i << "]"
                << j << "] = " <<
            test[i][j] << endl;
        }
    }
    return 0;
}
```

The output is shown in Figure 1.4.

```
Enter Integer :45
Enter Integer :12
Enter Integer :67
Enter Integer :78
Enter Integer :34
Enter Integer :78
Enter Integer :89
Enter Integer :23
Enter Integer :67

test[0][0] = 45
test[0][1] = 12
test[0][2] = 67
test[1][0] = 78
test[1][1] = 34
test[1][2] = 78
test[2][0] = 89
test[2][1] = 23
test[2][2] = 67
```

**Figure 1.4:** Output of the program to enter and display 2D array elements

### Applications of arrays

Arrays are essential because they allow many values to be stored in a single data structure while quickly accessing each value. This is possible as all values in the array are of the same type and, therefore, need the same amount of memory to store. Also, elements are kept in contiguous memory locations. To find where the element is kept, accessing an element in an array “y[i]” needs to find where the element is kept.

### Array operations

Operations with a one-dimensional array are as follows:

- Deletion – Involves deleting specified elements from an array
- Insertion – Used to insert an element at a specified position in an array

(c) Searching – An array element can be searched. The process of seeking specific elements in an array is called searching.

- Merging – The elements of two arrays are merged into a single one.
- Sorting – Arranging elements in a particular order, either in ascending or descending order.

### Limitations of using arrays

Several limitations must be carefully considered during the implementation of arrays. These includes:

- Arrays are of fixed size
- Inserting and deleting elements can be problematic because of the shifting of elements from their positions within a table row or column
- Data elements are stored in a contiguous memory location that may not always be available. To solve this implication of using arrays, you can use linked lists arrays.

### Advantage of arrays

The array has the following advantages:

- The array is simple and easy to use.
- It is also, faster to access the elements.

### Disadvantages of arrays

The array contains the following disadvantages:

- The cluster size should be known before utilising it
- All the components in the cluster must be adorned and put away in the memory

- (c) Embedding any element in the set needs moving of every element of its originals
- (d) Expanding the size of the cluster is a period taking cycle. It is not easy to grow the size of the exhibit at run time.

## Pointers

### Meaning of pointer

A pointer is a variable whose value is the address of another variable, that is, the direct address of the memory location. It is a user-defined data type that creates *variables* for holding the memory address of other variables. It is denoted by the “\*” operator. It represents storage space in memory (RAM) partitioned into a small portion called cells which are used to store values. Each cell has a unique address and occupies one byte in memory, whose address is always unassigned integers. If one variable carries the address of another variable, the first variable is said to be the pointer of another. Thus, a pointer is a variable whose value is also an address where each variable has two attributes: *address* and *value*, as shown in Figure 1.5.



**Figure 1.5:** The pointer and its variable

The syntax of a pointer is as follows:

```
type *ptr_name;
```

Where *type* is any data type of the pointer, and;  
*ptr\_name* is the pointer's name.

### Declaring pointer in C++

The pointer can be declared as follows:

```
Datatype*variable_name;
```

- (a) The *datatype* is the base type of the pointer, which has to be a valid C++ data type in the case of C++.
- (b) The *variable\_name* should be the name of the pointer variable.
- (c) An asterisk (\*) used above in the pointer declaration is the same as the asterisk used to perform multiplication operations. It marks the variable as a pointer.

The C++ uses the ‘&’ (reference) operator to return a variable’s address. For example, if *x* is a variable, &*x* returns the address of the variable. The following example shows the application of a pointer in a C++ program.

### Example 1: How to apply pointer in C++

```
#include<iostream>
using namespace std;
int main ()
{
    int y = 12;
    int *b; // declaration of pointer variable
    b = &y; // the address of y variable is assigned to a pointer b
    cout<<"Value at b = "<<b<<endl; // display address stored in pointer b
    cout<<"Value at variable y = "<<y<<endl; // display value stored in variable y
    cout<<"Value at *b = "<< *b; // display value stored in address of pointer b
}
```

The output of the program is shown in Figure 1.6:

```
Value at b = 0x6ffe04
Value at variable y = 12
Value at *b = 12
```

**Figure 1.6:** Output of a program to demonstrate application of pointers

**Note:** The data type of the pointer and variable should be the same.

### **Advantages of pointers**

The following are the advantages of pointers:

- (a) It can return multiple values from a function
- (b) It reduces the code and improves performance
- (c) It retrieves strings and trees
- (d) Used with arrays, structures, and functions
- (e) It allows you to access any memory location in the computer's memory
- (f) It can allocate and de-allocates memory dynamically

### **Disadvantages of pointers**

The following are the disadvantages of pointers:

- (a) If it is wrongly referenced, it will affect the entire program.
- (b) It may lead to a memory leak if the dynamically allocated memory is not freed.
- (c) An uninitialised pointer leads to a segmentation fault.

### **Application of pointers**

A pointer is used in various ways as follows:

1. It is used to access array elements.

An array of components can be accessed using pointers, as shown in example 2.

### **Example 2:** Accessing an array elements using C++ pointers

```
#include<iostream>
using namespace std;
int main() {
    int a[]={6,10,23,30};
    cout<<*(a +1);
    return 0;
}
```

## FOR ONLINE USE ONLY

The output for the program in example 2 is 10 is shown in Figure 1.7:

```
10
-----
Process exited after 0.3492 seconds with return value 0
Press any key to continue . . .
```

**Figure 1.7:** Output of a program to access array elements using pointers

2. It is used for dynamic memory allocation

**Example 3:** In C++, you can allocate dynamic memory as follows:

```
#include<iostream>
#include<stdlib.h>
using namespace std;
int main(){
    int i,*ptr;
    ptr =(int*) malloc(3*sizeof(int));
    if(ptr == NULL){
        cout<<"Error! memory not allocated.";
        exit(0);
    }
    *(ptr+0)=10;
    *(ptr+1)=20;
    *(ptr+2)=30;
    cout<<"Elements are:";
    for(i =0; i <3; i++){
        cout<<*(ptr + i);
    }
    free(ptr);
    return 0;
}
```

The output for the program in example 3 is “Elements are:10 20 30” is shown in Figure 1.8:

```
Elements are: 102030
-----
Process exited after 2.097 seconds with return value 0
Press any key to continue . . .
```

**Figure 1.8:** Output of a program for dynamic memory allocation using pointers

3. The pointers are used to pass arguments by reference in functions to increase efficiency.

Example 4, the program code in C++ to pass arguments by reference in functions as follows:

```
#include<iostream>
using namespace std;
void swap(int* x,int* y) {
    int t=*x;
    *x=*y;
    *y = t;
}
int main() {
    int r =3, s=9;
    swap(&r,&s);
    cout<< r<<s;
    return 0;
}
```

The output is 93 as shown in Figure 1.9

```
93
-----
Process exited after 1.157 seconds with return value 0
Press any key to continue . . .
```

**Figure 1.9:** Output of the program to pass arguments by reference

4. The pointer can be used to implement data structures like linked lists and trees as described in the next sections.

### **Pointer operations**

In C++, the pointer operators are of two kinds:

- (a) Reference operator (&): The reference operator (&) returns the variable's address, and
- (b) Dereference operator (\*): The dereference operator (\*) helps us to get the value that has been stored in a memory address.

## FOR ONLINE USE ONLY

For example, if you have a variable given the name num1, stored in the address 0x236 and storing the value 27, then

- (i) The reference operator (&) will return 0x236.
- (ii) The dereference operator (\*) will return 27.

Example 5: A C++ program to demonstrate pointer operators (reference and dereference operators)

Try these code in C++ and display the output;

```
#include <iostream>
using namespace std;
int main () {
    int x = 26;
    int *ip;
    ip = &x;
    cout << "Value of x is : ";
    cout << x << endl;
    cout << "Value of ip is : ";
    cout << ip << endl;
    cout << "Value of *ip is : ";
    cout << *ip << endl;
    return 0;
}
```

The output of the program shown in Figure 1.10:

```
Value of x is : 26
Value of ip is : 0x6ffe14
Value of *ip is : 26
```

**Figure 1.10:** Output of the program to demonstrate reference & dereference operators OR pointer operations

### Pointers in arrays

Arrays and pointers work based on a related idea. There are different things to consider when working with arrays having pointers. The array name denotes the base address of the array. This means that, when assigning an array's address to a pointer, do not use an ampersand (&).

#### For example:

**p = arr;** is correct since **arr** represents the arrays' address but

**p = &arr;** is incorrect.

You can implicitly convert an array into a pointer. For example:

```
int arr [20];
```

```
int * y;
```

This is a valid operation:

```
y = arr;
```

After the declaration, y and arr will be equivalent and share properties. However, a different address can be assigned to y, but you cannot assign anything to arr.

For example, you can traverse an array using pointers. Example of a C++ program to traverse the array given as arr[] = { 11, 45, 63, 26, 5, 24}.

```
#include <iostream>
using namespace std;
int main() {
    int *y;
    int arr[] = { 11, 45,
63, 26, 5, 24 };
    y = arr;
    for (int x = 0; x < 6;
x++) {
        cout << *y << endl;
        y++;
    }
    return 0;
}
```

The output of the program as shown in Figure 1.11

```
11
45
63
26
5
24
```

**Figure 1.11:** Output of a program to traverse array

## Records

Records are composite data types formed by several related items of different data types. This enables a programmer to refer

to these items using the same identifier, enabling a structured approach to use related items. A record will contain a fixed number of items. For example, a record for a book may include the title, author, publisher, number of pages, and whether it is literature or fiction.

Records store a collection of related data items, where all items have different data types. For example, you might set up a record called book, which stores the book's title, author name, and ISBN. Title and author are text, whereas the PublicationDate is set as a Date data type. You can write it as follows:

Book = Record

Title, Author As Text \* 50

ISBN As Text \* 13

PublicationDate As Date.

When the program runs, every time data is entered for the book, the user will type in up to 50 characters of text for the title, author, and up to 13 of text for ISBN. A variable may now be set up using this record data type, containing all of this data.

Generally, a record is a data structure that consists of a fixed number of variables called fields. Every field has an identifier (field name) and a data type. Each field in a record can have a different data type. This is very common in Spreadsheets and MS Access, and other databases. Some languages provide a built-in structure type that can be used to define a record.

For example, assume that you want to organise the individual student records such as registration number, name, sex, date of birth, average score, and grade into a single data structure.

*Student's\_regNo:* **integer**  
*Student's\_name:* **string**  
*Student's sex:* **string**  
*Student's\_average\_score:* **real**  
*Student's\_grade:* **char**

### **Structure of records**

Student's records can be tabulated as shown in Table 1.1

**Table 1.1:** Student's records

<b>sn</b>	<b>Student's_regNo</b>	<b>Student's_name</b>	<b>sex</b>	<b>Average_score</b>	<b>Grade</b>	Field
1.	7880	Maro Sambayeti	M	67	C	Records
2.	7565	Hadija Mkude	F	78	B	
3.	8665	Makala Ikande	M	87	A	
4.	7881	Bupe Mwakibete	F	91	A	

### **Application of records**

The following are the application of records:

- (a) It is used to store data or information in a database such as Microsoft Access.
- (b) It is used to group similar data.

### **Record operations**

The record operations include creating, accessing, and updating the field in the record.

#### *Creating a student's record*

The keyword 'record' is used to create records specified with the record name and fields. Its syntax is as follows:

```
record (recodname, {field1, field2, field3, field4 . . . . . fieldn})
```

The syntax to insert values into the record using C++ is as follows:

```
#recordname {fieldName1 = value1, fieldName2 = value2, fieldName3 = value3 . . . . . fieldNamex = valuex}
```

Example: A program to create a record using the C++ programming language.

```
#include <iostream>
#include <string>
using namespace std;
class student {
```

```
public:  
    string student_name;  
    int student_id;  
};  
int main(){  
    student S;  
    S.student_name = "Maro Sambayeti";  
    S.student_id = 7880;  
    return 0;  
}
```

Example: A program to access record values in the C++ programming language

```
#include<iostream>  
#include<string>  
using namespace std;  
class student {  
public:  
    string student_name;  
    int student_id;  
};  
int main(){  
    student S;  
    S.student_name="Maro Sambayeti";  
    S.student_id=7880;  
    cout<<S.student_id<<"\n"<<S.student_name;  
    return 0;  
}
```

Output is shown in Figure 1.12



```
7880  
Maro Sambayeti
```

**Figure 1.12:** Output of a program to access record values.

**Note:** Starting the number with zero (0) will make it octal; therefore, the program will display an error.

## Exercise 1.2

Answer the following questions:

1. Describe the static data structure.
2. Identify any four (4) advantages and disadvantages of static data structure.
3. With examples, describe the types of arrays.
4. Explain how to declare variables in an array. Give two examples.
5. What are the advantages and disadvantages of arrays? Describe any three for each.
6. Explain the applications of arrays.
7. Describe the array operations.
8. Give the meaning of the pointer.
9. Show how to declare a pointer in C++.
10. Point out the four advantages of a pointer.
11. Explain the application of pointers.
12. What is the meaning of records?
13. Suppose Juma wants to write a C++ program that uses 12 blocks of elements in memory, using the static data structure. He is then expected to store the other six more later. What would be the consequences of the program?

### User-defined data types

The user-defined data types allow a programmer to develop his/her data types and define what values program can take, for example, class, structure, union, and enumeration. These data types hold more complexity than pre-defined data types, but they can assist a programmer in reducing errors.

In computer science, the data types defined by the user are called derived datatypes or user-defined data types. In a programming language, user-defined data types are defined by the users in the program to suit their needs to store data either of the same or different types as per their requirements.

User defined data types are also known as Composite data types. They are called composite because they are derived from more than one built-in data type used to store complex data. These complex format data might contain tabular data, graphical data, and databases. For this purpose, we use complex data types called user-defined data types.

For every programming language, there are different types of user-defined data types. In C++, these user-defined data types are class, Union, structure, enumulation, and typedef.

(a) **Class:** This is a user-defined data type that holds data members and functions whose access can be specified as private, public, or protected. It uses the '**class**' keyword for defining the data structure.

(b) **Structure:** A structured data type groups data items of different types into a single type. For example, a structure can be an address, which contains information such as block number, plot number, building name, street, city, country and pin code. The keyword "struct" is used to define this.

(c) **Union:** The Union is a type of data structure where all the members of that Union share the same memory location. If any changes are made in the Union, they will also be visible to others. The 'union' keyword is used to define this user-defined data type.

(d) **Enumeration:** It helps to assign names to integer constants in the program. The keyword ‘enum’ is used. It is used to increase the readability of the code.

(e) **Typedef:** This defines a new name for an existing data type. It does not create a new data class. It makes code readability easy and gives more clarity to the user.

### *Application of user-defined data structure*

- User-defined data types are the building blocks for other data types that model the structure and behaviour of data.
- They are used to implement mathematical vectors and matrices.
- They are used to model sets or collections in computer programming.
- It creates other data structures such as, lists, stacks and queues.

### *The structure*

A structure is a collection of multiple variables of different data types grouped under a single name for convenient handling. The members of structures can be ordinary variables, pointers, arrays or even another structure.

### *Declaring structures*

The structures are created by using the `struct` keyword

```
struct
{
    member a;
    member b;
    ....;
    member n;
};
```

For example, you can create a structure to store the student’s marks

```
struct student //  
student as structure name  
{  
    //structure members  
    int stdid;  
    char stdname[40];  
    float marks, attendance;  
};
```

**Example:** The C++ program to demonstrate structures in C++

```
// C++ program to demonstrate  
// Structures in C++  
#include <iostream>  
using namespace std;  
struct Point {  
    int x, y;  
};  
int main()  
{  
    // Create an array of  
    // structures  
    struct Point arr[20];  
    // Access array members  
    arr[0].x = 20;  
    arr[0].y = 50;  
    cout << arr[0].x << ", "  
    << arr[0].y;  
    return 0;  
}
```

The program’s output is 20, 50 as shown in Figure 1.13.

20, 50

**Figure 1.13:** The output of the program to demonstrate structures in C++

## Union

The Union is a user-defined data type in which all members share the same memory place. For example, in the following C++ program, y and z share the same location. If we change z, you can see the changes reflected in y.

```
#include <iostream>
using namespace std;
// Declaration of union is
// the same as the structures
union test {
    int y, z;
};
int main()
{
    // A union variable t
    union test t;
    // t.z also gets value 4
    t.z = 4;
    cout << "After making z = 4:" 
        << endl
        << "y = " << t.y
        << ", z = " << t.z
        << endl;
    // t.y is also updated to 20
    t.z = 20;
    cout << "After making z = 20:" 
        << endl
        << "y = " << t.y
        << ", z = " << t.z
        << endl;
    return 0;
}
```

The program's output is as shown in Figure 1.14:

```
After making z = 4:
y = 4, z = 4
After making z = 20:
y = 20, z = 20
```

**Figure 1.14:** The output of a program to demonstrate union in C++

In C++, an enumeration (or enum) is primarily used to give integral constant names, making the program easier to comprehend and maintain. In enumeration, if you do not provide the integral values explicitly to the strings, then, in that case, the strings automatically start assigning the integral values starting from value 0, the same as the case of 0-based indexing.

Points to remember for C++ Enum

- enum improves type safety
- enum can be easily used in switch
- enum can be traversed
- enum can have fields, constructors and methods
- enum may implement many interfaces but cannot extend any class because it internally extends the Enum class

## Example:

```
enum result {pass = 60, fail = 16};
```

Here, we have given the integer value 60 to be pass and 16 as fail; therefore, if we write,

```
enum result res;
```

```
res = pass;
```

Then, the value of res would automatically be 60.

## Typedef

The keyword `typedef` in C++ allows you to define new data type names explicitly. The use of `typedef` does not create a new data class but establishes a name for an existing type. This can improve a program's portability (the ability of a program to be used across different

types of machines, i.e., mini, mainframe and micro without requiring significant changes to the code) because only the `typedef` statements would need to be changed. By allowing descriptive names for standard data types, `typedef` can also help with self-documenting code.

### Syntax:

`Typedef type name;`

An example C++ program to demonstrate `typedef` user-defined datatype

```
#include <iostream>
using namespace std;
// After this line, BYTE
can be used
// in place of unsigned char
typedef unsigned char BYTE;
int main()
{
    BYTE c1, c2;
    c1 = "d";
    cout << " " << c1;
    return 0;
}
```

### Output:

d

#### Benefits of user-defined data types

- User-defined data types store data elements of either the same or different types. This gives more flexibility for the programmer to store different data types in a single variable as per their needs and requirements.

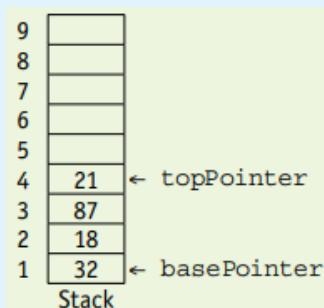
(b) Reusability: Once defined, these data types can be reused within many definitions, saving coding time.

- Flexibility: They allow us to create a data structure per our requirements and needs.
- Encapsulation: In Java/Python, the variables and data values stored in user-defined data types are hidden from other classes as per their accessibility declaration, i.e. public, private, and protected. The data values remain hidden and safe.



#### Activity 1.1: Differentiating types of data structures

- In pairs or groups of three, identify at least three uses for a 1 dimension array and three uses for a 2-dimensional array.
- Compare array structures with record structures, and decide if any of your uses would be better structured as records.
- Show the stack, and the value of the top pointer and base pointer when an item popped (taken out) off the stack and 54 followed by 70 pushed onto the stack.



**Exercise 1.3**

1. What do you understand about the user-defined types?
2. Describe the five types of user-defined data types.
3. Describe the structured type.

**Project:**

Develop a simple C++ program that includes the struct, union, and enum concepts. The program should address any problem in your school or community.

**Dynamic data structure*****Meaning of dynamic data structure***

The dynamic data structure is the type of data structure in which the size of the structure is not fixed and can be modified when the performance of the operations is carried on them. The dynamic data structures are designed to facilitate the change of data structures in the run time. The size can be randomly updated during the run time, which may be considered efficient concerning the memory complexity of the code. Examples of dynamic data structures are linked lists, stacks and queues.

**Advantages of dynamic data structure**

The following are the advantages of dynamic data structure:

- (a) It makes the most efficient use of memory since the data structure only uses as much memory as it needs.
- (b) The memory allocation is not fixed, so there will be no problem adding and removing data items.

- (c) Easier to program as there is no need to check data structure size.

**Disadvantages of dynamic data structure**

- (a) Because the memory allocation is dynamic, it is likely for the structure to ‘overflow’ should it exceed its allowed limit. It can also ‘underflow’ should it become empty.
- (b) It is complicated to program as the software needs to keep track of its size and data item locations.
- (c) It can be very inefficient because the memory for the data structure is defined regardless of whether it is needed or not when the program is executing.

**Types of dynamic data structure**

Dynamic data structures are linked lists, arrays, and queues. Each of these types is discussed in the following subsections.

**Linked lists**

The linked list is a dynamic data structure, among others, that can grow and shrink during the execution of a program. The associated pointer would give the memory location of the next data item in the linked list.

***Meaning of linked lists***

A linked list is a sequence of data structures connected via links. A linked list is a sequence of links that contain items. Each link has a connection to another link. The linked list is the second most-used data structure after the array. The linked list can be defined as a collection of objects called nodes randomly stored in the memory. A node

contains two fields, the first, which stores data at that particular address and the second is the pointer, which has the address of the next node in the memory. The last node of the list contains a pointer to the null.

The following are essential terms to understand the concept of a linked list.

**Link** – Each link of a linked list can store data called an element.

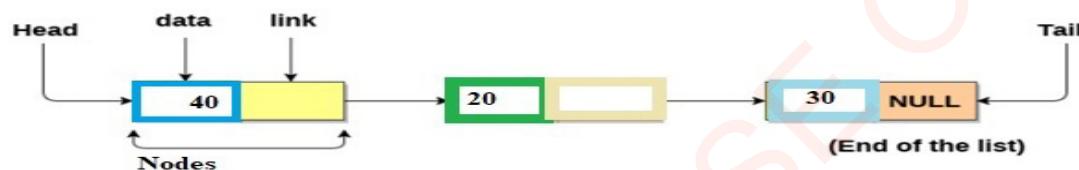
**Next** – Each link of a linked list contains a link to the following link called Next.

**Linked list** – A linked list contains the connection link to the first link called First.

**Head** – Is the first node and tail is the last node.

### **Structure of linked lists**

The structure of the linked list comprises two parts in each node. These parts are; the one which represents data and that contain the address of the next node, as shown in Figure 1.15.



**Figure 1.15: Linked list**

### **Advantages and disadvantages of linked list**

Linked lists have both advantages and disadvantages over other data structures.

#### **Advantages of linked lists**

The advantages of linked lists include:

- Easy insertion and deletion of elements: Insertion and deletion are efficient and take constant time as the links are manipulated hence, there is no need to give the initial size of the list.
- No memory wastage: the size of the linked list increase and decreases during runtime. This utilises the memory efficiently, hence, no wastage of memory.
- They can be expanded constantly: The linked list can grow and shrink depending on the data to be inserted. With a linked list, you can start with space for just one allocated element and then add on new elements easily without the need to do any copying and reallocating
- Implementation: The implementation of stack and queue is very easy using a linked list.
- Easy of moving pointer: In the linked list, it is very easy to insert and delete a node. After insertion and deletion, there is no need to shift every element; instead, it needs only to update the address in the next pointer.

***Disadvantages of linked lists***

The following are the disadvantages of a linked list:

- (a) Memory usage: The node in a linked list occupies more memory than the array as each node occupies two types of variables, i.e., one is a simple variable, and another is a pointer variable that occupies 4 bytes in the memory.
- (b) Traversal: In a linked list, traversal is not easy. If you want to access the element in a linked list, you cannot access the element randomly, but in the case of an array, you can randomly access the element by index. For example, if you want to access the 3<sup>rd</sup> node, you must traverse all the nodes before it. So, the time required to access a particular node is extensive.
- (c) Reverse traversing: Backtracking or reverse traversing is difficult in a linked list. It is easier but requires more memory to store the back pointer.

***Application of linked lists***

The following are some applications of linked lists:

- (a) Implement dynamic memory management functions of the operating system.
- (b) A circular linked list is used to implement operating system or application functions requiring round-robin tasks execution.
- (c) The linked list is used in a circular queue to maintain multiple players' playing sequences in a game.

- (d) Implement stacks, queues, binary trees and graphs of pre-defined size.
- (e) The circular linked list is used in a slide show where a user wants to go back to the first slide after displaying the last one.
- (f) A linked list is used for the polynomial implementation of mathematical operations.
- (g) A doubly linked list is used to implement forward and backward buttons in a browser to move backwards and forward in the opened pages of a website.
- (h) It is used when a user selects the desired application by pressing the **Alt+tab** key combination to navigate through the opened application.

***Types of linked list***

Linked lists can be divided into three types:

- (a) A Singly-linked list is a linked list traversed in one direction from head to tail.
- (b) The Doubly linked list is a type of linked list traversed in both directions, front and back.
- (c) A Circular linked list is a list in which the last node points to the first node.

***Linked lists operations***

The primitive operations performed on the linked list are as follows:

- (a) Creation
- (b) Insertion
- (c) Deletion
- (d) Traversing
- (e) Searching
- (f) Concatenation

**Creation operation:** This is used to create a linked list. Once a linked list is created with one node, the insertion operation can add more elements to a node.

**Insertion operation:** Is used to insert a new node at any specified location in the linked list. A new node may be inserted;

- At the beginning of the linked list
- At the end of the linked list
- At any specified position in between in a linked list. For example, Figure 1.16 shows the insertion operation of inserting a new node at the beginning of an existing linked list where,
  - Show an existing linked list of the region in both Tanzania mainland and Zanzibar island.
  - Shows the insertion of a new region called Tabora, and
  - Shows a resulting new liked list after insertion operation.

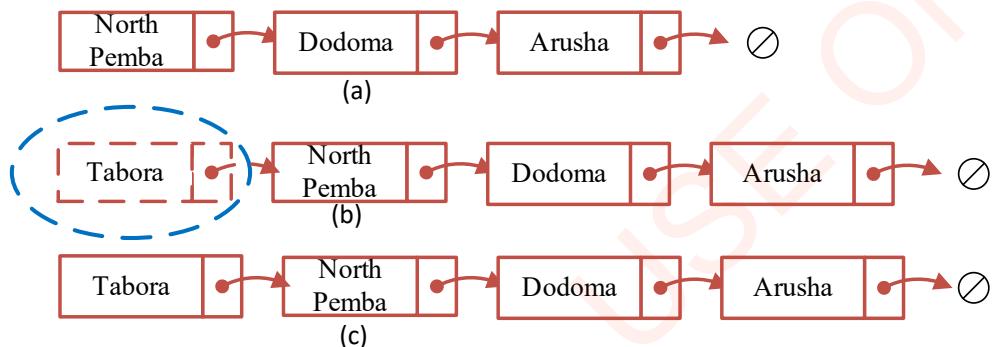


Figure 1.16: Insertion operation

**The deletion operation:** This is used to delete an item (a node) from the linked list. A node may be deleted from the;

- Beginning of a linked list
- End of a linked list and
- Specified location of the linked list. (See Figure 1.17).

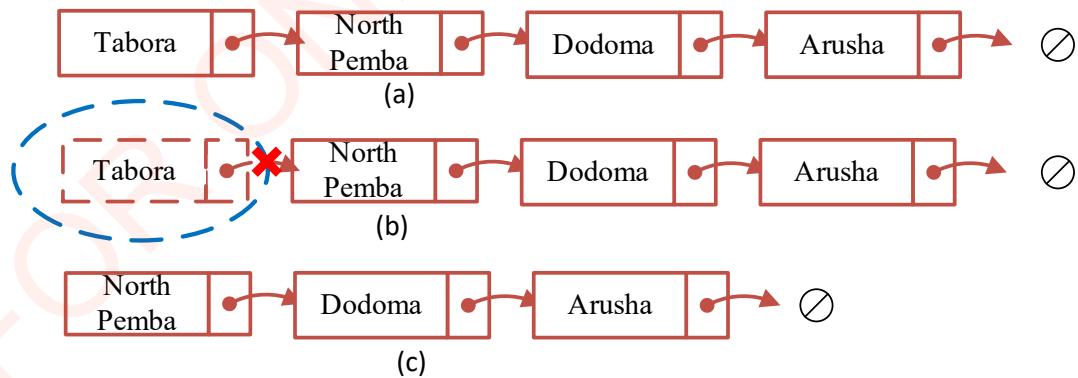
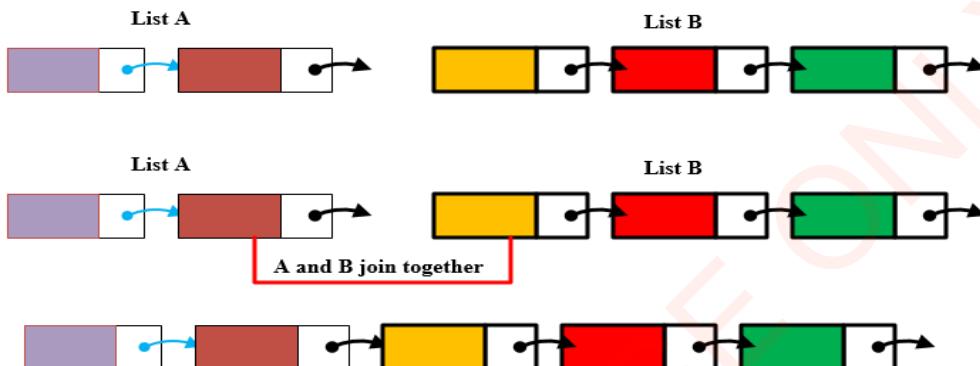


Figure 1.17: Deletion operation

**Traversing:** This is the process of going through all the nodes from one end to another end of a linked list. You can visit from left to right in a singly linked list, forward traversing, nodes only. In a doubly-linked list, forward and backward traversing is also possible.

**Concatenation:** This is the process of appending the second list to the end of the first list.

For example, consider a list A having x nodes and B with y nodes; then the operation concatenation will place the 1<sup>st</sup> node of B in the (x+1)<sup>th</sup> node in A. After concatenation, A will contain (x+y) nodes. (Figure 1.18)



**Figure 1.18:** Concatenation in a linked list

## Stacks

### Meaning of stacks

A stack is an Abstract Data Type (ADT) and is a linear data structure that can be accessed only at one of its ends for storing and retrieving data. It is analogous to a stack of trays in a cafeteria whereby, new trays are put on the top of the stack and taken out the top. The last tray placed on the stack is the first tray removed from the stack and based on this principle, a stack is called a Last-in-first-out (LIFO) structure. Here, the element placed (inserted or added) last is accessed first. In stack terminology, the insertion operation is called the PUSH operation and the removal operation is called the POP operation.

A tray can only be taken if there is at least one tray on the stack, and a tray can be added to the stack only if there is enough room when the stack is not too high. Therefore, a stack contains operations that change its status and operations that check this status. The operations are as follows;

**clear()**—Clear the stack.

**IsEmpty()**—determines whether the stack is empty. It returns 1 (true) if the stack is empty otherwise, it returns 0 (false).

**push()**—Put the element on the top of the stack.

**pop()**—Take the topmost element from the stack.

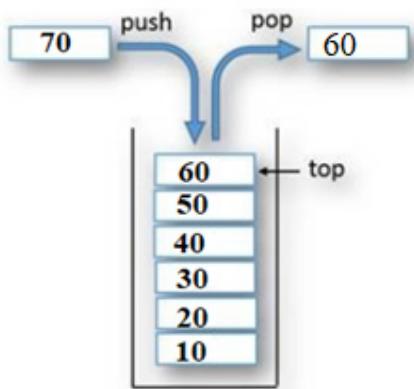
**topEl()**—Return the topmost element in the stack without removing it.

**Size()**: Returns the number of elements in the stack

A real-life example of a stack is a deck of cards or a pile of plates.

### Structure of stacks

The stack can be demonstrated as in Figure 1.19.



**Figure 1. 19:** The stack structure and its operation

### Application of stacks

Generally, the stack is beneficial when data are stored and retrieved in reverse order. One application of the stack is in matching delimiters in a program. This is

an example because delimiter matching is part of any compiler: No program is considered correct if the delimiters are mismatched. In C++ programs, we have the following delimiters: parentheses “(” and “)”, square brackets “[” and “]”, curly brackets “{” and “}”, and comment delimiters “/\*” and “ \*/”. It is also used to undo, redo, and navigation bars.

### Stacks operations

#### PUSH operation

The mechanism of PUSH operation in a stack is conducted as follows:

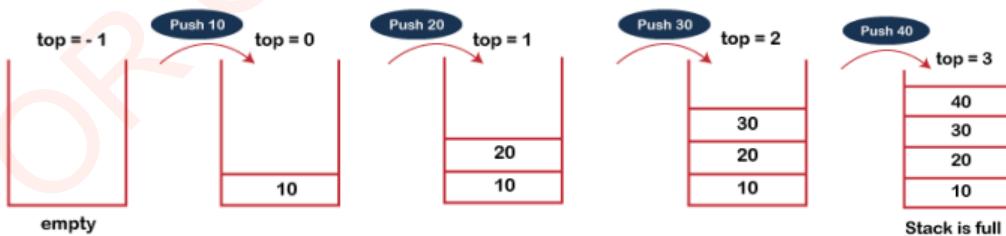
Before embedding a component in a stack, you check whether the stack is complete.

If you attempt to embed the component in a stack, and the stack is full, at that point, the flood condition happens.

When you introduce a stack, set the maximum estimation as - 1 to watch that the stack is unfilled.

When the new component is pushed into the stack, first, the size or number of the top gets increased, i.e.,  $\text{top}=\text{top}+1$ , and the component will be put at the new position of the top.

The components will be embedded until we arrive at the maximum size of the stack (i.e.  $\text{top} = 3$ ), as shown in Figure 1.20.



**Figure 1.20:** Push operation

**POP operation**

The mechanism for POP operation in a stack is conducted as follows:

- Before erasing the component from the stack, you check whether the stack is vacant.
- If you attempt to erase the component from the empty stack, the sub-current condition happens.
- If the stack is not unfilled, you first access the component, which is pointed at the top.
- Once the pop activity is played out, the top is decremented by 1, i.e.,  $\text{top}=\text{top}-1$ .

**Activity 1.2: Demonstration of the stack****Resources:**

Abacus, bottle caps, two pieces of spokes, a piece of wood

**Procedure:**

Take an Abacus or make it using wood and a piece of the spoke. Fix the pieces of spokes into a piece of wood to create an Abacus

Label the first rod "A" and the second rod "B." Take five bottle caps and put them into the first rod of an Abacus. Take the other three bottle caps and the first rod one after another. Take out three bottle caps from rod A to B in the abacus

Observe and write on the following:

- Tell from which point the bottle caps were added.
- Do you think you can remove the second cap from rod A without removing others on its top? Give reasons.

(iii) Can you add a cap from the bottom of the fixed rod? Give reasons.

(iv) What types of data structure does the activity represent?

Draw the diagram to demonstrate the activity of adding and removing the bottle caps to and from the rods in the abacus.

Can you give one example in your daily life which exhibits the same principle in this activity?

**Think about the following:**

The vehicles on weigh bridge or at the narrow one-way bridge; then the phone calling and answering system. Afterwards, think about the basis on which these incidents are being attended. Is it on a last -in first-served or first-in-first-served basis? Lastly, think about what will happen to other vehicles if one vehicle in the service fails.

**Queues*****Meaning of queue***

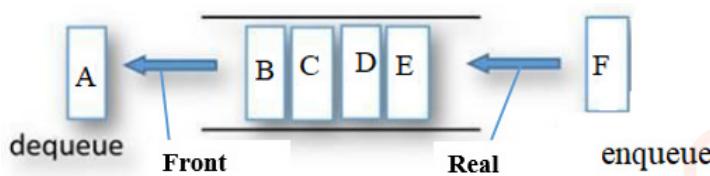
The queue is an abstract data structure, somewhat similar to stacks. Unlike stacks, a queue is open at both ends. The queue follows the First-In-First-Out methodology, i.e., the data item stored first will be accessed first.

Queues are what you usually refer to as lines. Its essential features are, that it is ordered and that access is restricted to its ends: things can enter a queue only at the rear and leave the queue only at the front. The queue holds a sequence of elements that allows insertions (enqueue) only at

one end, called the back or rear, and deletions (dequeue) and access to elements at the other end, called the front. In computing, queues are essential because resources provide service on a first-come-first-served basis, such as jobs sent to a printer or processes waiting for the CPU in an operating system.

A real-world example of a queue can be a one-way road or single-lane bridge, where the vehicle first enters and exits. More real-world examples are queues at the ticket windows and bus stops, banks, queuing for voting, taking food, and hospitals. Figure 1.21 shows the structure of the queues.

### *Structure of queues*



**Figure 1.21:** Structure of the queue

### *Application of queues*

The real-life queue is realised in what you all participate in from time to time. The following are some of the applications that use queues:

- You wait in line at railway and bus stations for reservation counters and tickets.
- You wait in the dining halls and cafeteria line.
- You wait in a queue in the hospital when you want to present your illness case to a doctor.
- You issue a print command from our office to a shared networked printer; the print task is lined up in a printer queue. The print command issued first will be printed before the subsequent commands in the line (in the queue).
- An operating system is also using different queues to control process scheduling. Processes are added to the processing queue and used by an operating system for various scheduling algorithms.
- The first-served scenario requires a queue to simulate real-world queues such as lines at a bank counter or any other first-come.

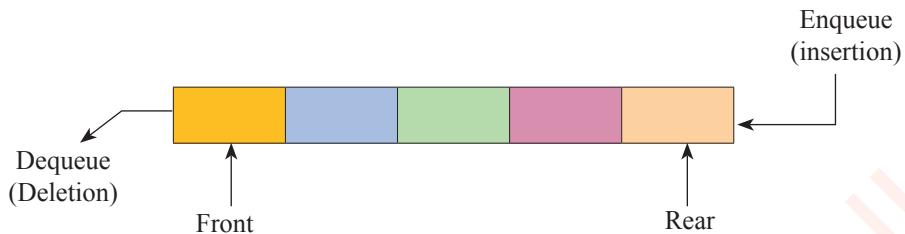
### *Queues operations*

The queues are a homogeneous collection of elements in which new elements are added at one end, called the rear, and the existing elements are deleted from the other end, called the front. The basic operations that can be performed on a queue are:

- Insert (or add) an element to the queue (enqueue)

- (b) Delete (or remove) an element from a queue (dequeue)

Enqueue operation will insert (or add) an element to the queue at the rear end by incrementing the array index. The dequeue operation will delete (or remove) from the front end by decrementing the array index and will allocate the deleted value to a variable. When implemented using arrays, the total number of elements present in the queue is  $\text{front} - \text{rear} + 1$ . Figure 1.22 illustrates the basic operations of the queue.



**Figure 1.22: Queues operation**

When deleting or removing an element from the queue, underflow occurs. It is impossible to delete or remove any element when no element exists in the queue. Suppose the maximum size of the queue when implemented using arrays is 20; if we try to enqueue, i.e., insert or add an element to the queue, overflow occurs. It is naturally impossible to insert more elements when the queue is full.

### Non-linear data structure

#### Tree

A tree is a non-linear data structure that is an abstract and typically hierarchical structure consisting of nodes with a parent-child relationship. It is used in organisation charts, programming environments, and file systems. There are four basic things connected with any tree: the distinction between nodes,

orientation structure, the value of nodes, and the number of levels. Examples of tree data structures include binary, B-tree, heap, and splay.

#### Binary tree

##### Meaning of binary tree

A binary tree is a tree for which each internal node is allowed to have at most two child nodes. The children of an internal node are called the left child and the right child, depending on the relationship with the parent. The recursive definition of a binary tree is either a tree consisting of a single node or a tree whose root has an ordered pair of children, each of which is a binary tree. The applications of binary trees represent arithmetic expressions, create decision processes and search. Figure 1.23 illustrates the structure of the binary tree.

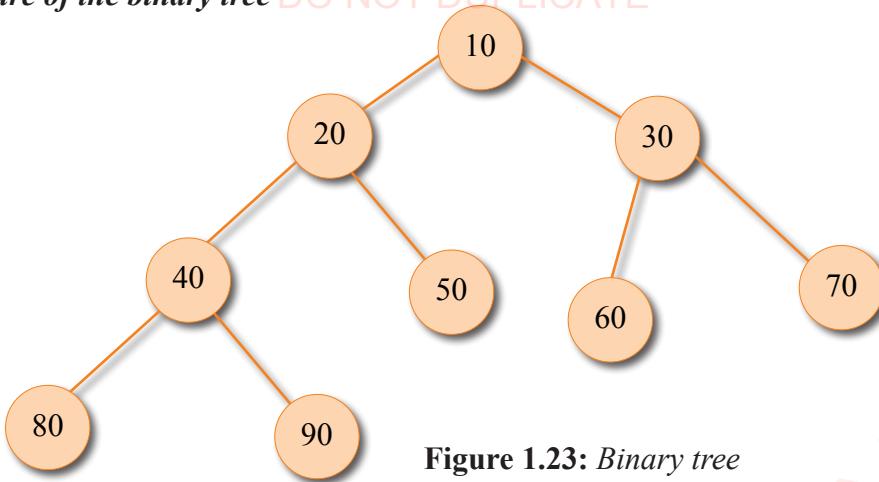


Figure 1.23: Binary tree

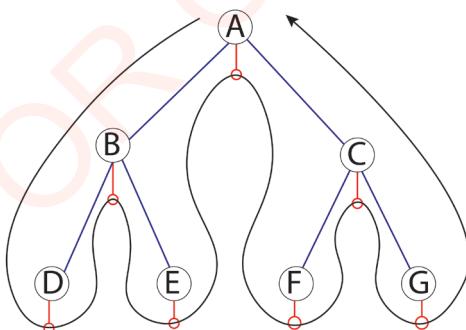
### **Binary tree operations**

**Preorder Traversal**— A node is visited before its descendants (in the example, the Preorder traversal sequence is 10,20,40,80,90,50,30,60,70).

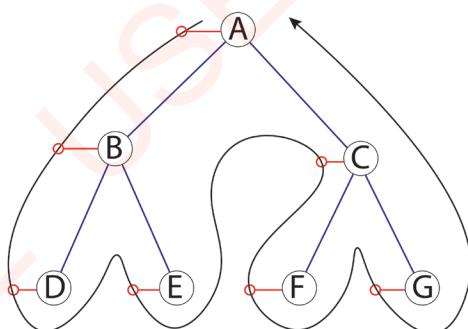
**Inorder Traversal**— A node is visited between traversal of its descendants in the example; the Inorder traversal sequence is 80,40,90,20,50,10,60,30,70.

**Postorder Traversal**— A node is visited after its descendants (in the example, the Postorder traversal sequence is 80,90,40,50,20,60,70,30,10).

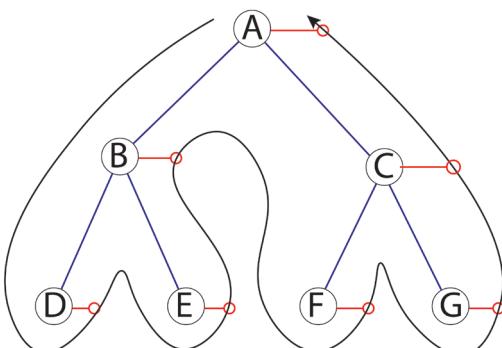
Figure 1.24 shows the binary tree traversing which will help you to understand the binary tree traversal quickly.



Inorder D, B, E, A, F, C, G



Preorder A, B, D, E, C, F, G



Postorder D, E, B, F, G, C, A

Figure 1.24: The binary tree traversing

**Note:** If  $n$  is the number of nodes,  $x$  is the number of external nodes,  $i$  is the number of internal nodes, and  $h$  is the height of the binary tree, then the following are the essential properties of binary trees:

$$x = i + 1$$

$$n = 2*x - 1$$

$$h \geq \log_2 x$$

$$h \geq \log_2(n+1)-1$$

#### Inorder operation in a binary tree

- (i) Traverse the left subtree, i.e., call Inorder (left subtree)
- (ii) Visit the root
- (iii) Traverse the right subtree, i.e., call Inorder (right-subtree)

#### Uses of Inorder

For example, in binary search trees (BST), Inorder traversal gives nodes in non-decreasing order. To get nodes of BST in non-increasing order, a variation of Inorder traversal can be reversed.

For example, Inorder traversal for Figure 1.25 given tree is 14, 26, 58, 10, 30.

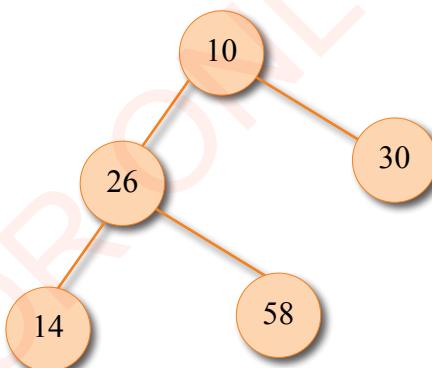


Figure 1.25: The Inorder traversal

#### Pre-order operation in a binary tree

Algorithm for Preorder traversal:

- (i) Visit the root.
- (ii) Traverse the left subtree, i.e., call Preorder (left-subtree)
- (iii) Traverse the right subtree, i.e., call Preorder (right-subtree)

#### Uses of Preorder

- (i) Preorder traversal is used to create a copy of the tree.
- (ii) Preorder traversal is also used to get prefix expressions on an expression tree.

**Example:** The Preorder traversal for the given Figure 1.26 is 11 23 41 15 13.

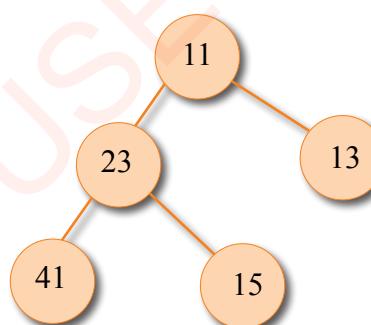


Figure 1.26: Preorder traversal

**Example 1:** Study the input in Figure 1.27. Then, write its corresponding output by using pre-order traversal.

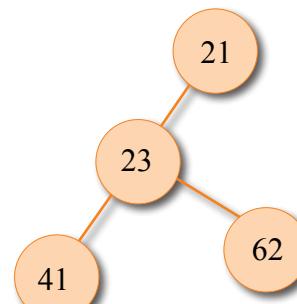


Figure 1.27: Pre-order traversal

Output: 21 23 41 62

**Post-order**

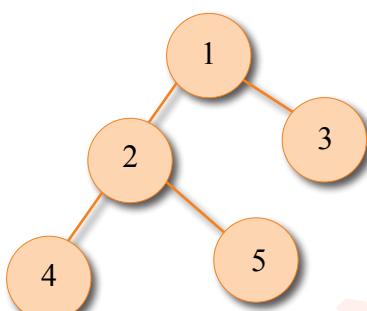
Algorithm for Postorder (binary tree)

1. Traverse the left subtree, i.e., call Postorder (left-subtree)
2. Traverse the right subtree, i.e., call Postorder (right-subtree)
3. Visit the root.

**Uses of Postorder**

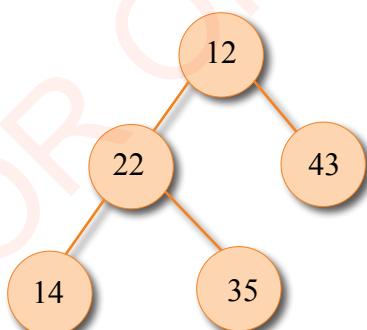
- (i) Postorder traversal is used to delete the tree.
- (ii) Postorder traversal is also helpful for getting the postfix expression of an expression tree.

**Example 2:** Postorder traversal for the given Figure 1.28. is 4 5 2 3 1.



**Figure 1.28:** The postorder traversal

**Example 3:** Consider input as given in Figure 1.29 and its corresponding output using postorder traversal:



**Figure 1.29:** Preorder traversal

Output: 14, 35, 22, 43, 12

*Application of binary tree*

The following are the applications of binary tree:

- (a) The binary tree is used to implement indexing of the Segmented Database.
- (b) It is used in implementing efficient cache in hardware and software systems.
- (c) A binary Search Tree is used to search elements efficiently and as a collision handling technique in hash map implementations.
- (d) The binary tree is the primary data structure in Microsoft Excel and spreadsheets.
- (e) A binary Heap (Binary Tree variant of the heap) is used to implement a Priority Queue efficiently, which is used in Heap Sort Algorithm.
- (f) A binary tree with nodes as operations is used to compute arithmetic expressions in some compilers.
- (g) A balanced Binary Search Tree is used to represent memory to enable fast memory allocation.
- (h) A self-balancing Binary Search Tree is used in implementing games to model a faulty search process
- (i) Binary trees with 2 child is used to represent routing data that fluctuates efficient traversal.

**Algorithms****Concept of algorithm**

One of the significant steps to solving any problem is to analyse the problem.

## FOR ONLINE USE ONLY

Analysing a problem involves understanding the problem and its requirement. Problem requirements include users, input data, output data and general system functionality. For example, when one wants to develop a school management system, the requirements might be the number of students, teachers, and the school administrator who will be responsible for updating data. After analysing a problem, the next step is to design an algorithm.

An algorithm involves step-by-step analysed procedures to solve a particular problem. It is a set of instructions written to accomplish the defined task. The algorithm is not a program, so you do not need to use a programming language to design it. It is just a way of solving a problem systematically, which can be represented by either a flowchart or pseudocode. The algorithm should define the beginning and end of the problem with a specific number of steps. The steps of the algorithm normally involve decisions, including logic and comparison. All programs designed within the computer system are the implementation of the algorithm. Algorithms are used for calculation, data processing and automatic reasoning. However, the concept of the algorithm can be applied even in solving typical daily life tasks, such as cooking food, studying, teaching, and travelling.

### Importance of algorithm

Through an algorithm;

- (a) it is possible to know if there is a possibility of solving a particular problem or not. This includes

implementing the given problem requirement to solve a problem.

- (b) The fastness or the slowness of the designed system is measured.
- (c) We can solve the whole problem at a time or divide it into manageable parts.

### Characteristics of an algorithm

The algorithm to solve the required problem must be correctly and well-defined. Thus must obey the following characteristics:

- (a) **Finiteness:** An algorithm should have a defined number of steps that must terminate at the end of the steps.
- (b) **Definiteness:** Every step in the algorithm should be defined in one meaning. The purpose of being conducted should be clear with no ambiguity.
- (c) **Effectiveness:** Any operation within the algorithm must be performed efficiently and follow the same principle required to design the algorithm.
- (d) **Well-defined inputs:** The quantities used as the input must be well-defined and specified.
- (e) **Well-defined output:** An algorithm must have well-defined results and should match with the desired output.

### Advantages of algorithms

- (a) It is easy to understand because it uses normal language in writings
- (b) It is the step-by-step representation of a solution to a given problem

- (c) The problem is broken down into smaller steps for easier program writing

### Disadvantages of Algorithms

- (a) Writing an algorithm needs time to analyse the problem, which is time-consuming.
- (b) It is difficult to write high-order statements like branching and looping in an algorithm.

### How to write an Algorithm

As you have seen, an algorithm is a basic tool and very useful for understanding a problem through its step-by-step analysis. However, there are no well-defined standards for writing algorithms; instead, it depends on the problem and available resources. All programming languages share basic code constructs like do, for, while, and if-else. With this, you note that an algorithm is not written to support a particular programming code, but these common constructs are used to write an algorithm clearly with some steps as follows:

- Determine the input and output of the problem
- Find the correct data structure to present the problem
- Try to reduce the problem to a variation of a well-known one
- Decide whether you look for a recursive or imperative one or a mixed algorithm
- Write an algorithm

### Examples of the algorithm:

**Example 1:** Algorithm for multiplication of two numbers

Step 1: Start

Step 2: Enter the first number

Step 3: Enter the second number

Step 4: Multiply two numbers and store the result

Step 5: Display the result

Step 6: Stop

**Example 2:** Algorithm to find the largest number between two numbers

Step 1: Start

Step 2: Input two numbers, A and B

Step 3: Compare if A is greater than B; if yes, go to step 4; otherwise, go to step 5

Step 4: Print A is greater than B

Step 5: Compare if B is greater than A; if yes, go to step 6; otherwise, go to step 7

Step 6: Print B is greater than A

Step 7: Stop.

**Example 3:** Write an algorithm to find the biggest of two numbers

Step 1: Start

Step 2: Read two numbers, say A and B

Step 3: If  $A > B$ , then BIG = A, otherwise BIG = B

Step 4: Write BIG

Step 5: End

**Example 4:** Write an algorithm to find the Area of the Square.

Step 1: Start

Step 2: Input Side Length of Square says L

Step 3: Area =  $L \times L$

Step 4: Display AREA,

Step 5: End

**Example 5:** Write an algorithm to find Simple Interest.

Step 1: Start

Step 2: Input value of Principle amount (P), time in years (N), and % interest (R)

Step 3: Calculate SI =  $(P \times N \times R) / 100.0$

Step 4: Display Simple Interest (SI)

Step 6: End

**Example 6:** Write an algorithm to find compound interest.

Step 1: Start

Step 2: Input value of principle Amount (P), Time in Years (N), and % Annual Rate of Interest (R)

Step 3: Calculate CI =  $P (1+R/100)^N - P$

Step4: Display Compound Interest (CI)

Step6: End

**Note:** Two people can use different algorithms in solving similar problems as in the case of Example 2 and Example 3.

### Types of algorithms

In algorithms, you use to put similar approaches in solving a problem in the same group so that we can classify them one type of algorithm after another in which a problem can be analysed briefly and easily find a solution. There are many algorithms, but this book deals with two fundamental types, recursive and iterative.

### Recursive algorithms

This is an algorithm that calls itself repeatedly with a smaller value as inputs generated after solving the current inputs until the problem is solved. This means a process frequently uses the first

inputs as a requirement of the second step. Suppose you want to arrange three books labelled A, B, C, the following are the possible ways: ABC, ACB, BAC, BCA, CAB and CBA. This means there are six ways of arranging any three objects. Mathematically, that scenario can be solved by using the concept of factorial of n-object, for example, a  $3! = 3 \times 2 \times 1 = 6$  ways. This signifies that if five students are sitting on the horizontal bench, the ways of arranging them will be  $5! = 5 \times 4 \times 3 \times 2 \times 1 = 120$  ways.

Generally, methods of organising any given n-objects will be as;

$$n! = n \times (n - 1) \times (n - 2) \times (n - 3) \times (n - 4) \times \dots \times 2 \times 1.$$

This system of breaking a problem into more minor problems using the solution applied to solve the main problem is called recursion.

The algorithm that applies recursion to solve the problem is known as a recursive algorithm. Recursive algorithms solve a problem by breaking it into a simple problem using the same method by repeatedly calling themselves.

### Implementation of a recursive algorithm

You can use a recursive function to implement recursive algorithms. The recursive function contains a statement that causes the same function to execute again before completing the current call. It is the function that calls itself until the given condition is met. The result from one execution in a recursive function is used as the input of the next recursive. The following example represents a C++ program code to demonstrate the recursive function:

```

int factorial (int n) {
if(n==0)
return 1;
else
return(n*factorial(n-1));
}
From the given
demonstration code above;
    factorial      (3)      =
3*factorial(2)
    factorial      (2)      =
2*factorial(1)
    factorial(1) = 1* factorial
        (0)
    factorial (0) =1,
Then,
    factorial(3) = 3*2*1*1
∴ factorial(3) = 6.

```

The given solution explains that a problem is split into more minor instances of the same problem.

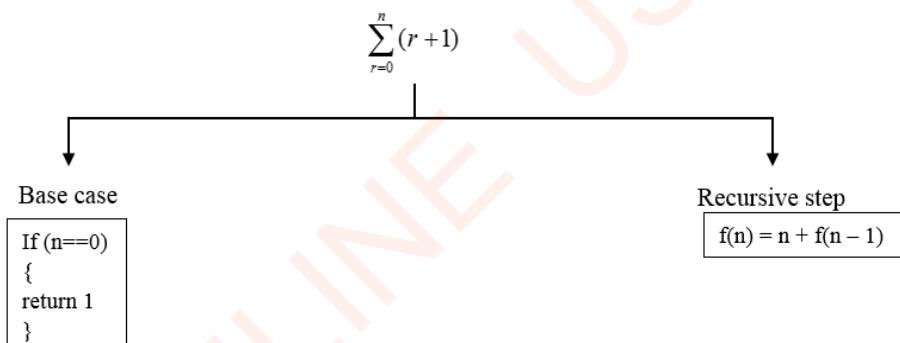
**Note:** The recursive function divides a problem into the base case and the recursive step. The base case is the most straightforward instance of the problem that stops the recursion function. It provides the result of the function when the condition is met. Also, the recursive step is responsible for providing solutions by making recursive calls and decreasing the size of inputs.

Consider some of the positive integers:

$$1 + 2 + 3 + 4 + 5 + \dots + n + 1 =$$

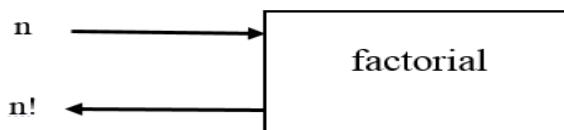
$$\sum_{r=0}^n (r+1)$$

Where:



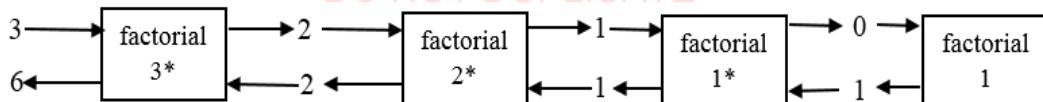
**Note:** If there is no base case in the recursive function, the recursive function will continue to repeat.

Recursive execution can be represented by using a data flow model. The box represents procedure invocation through data flow into and out of the box shown in Figure 1.30. Figure 1.31 represents the recursive data flow model and the use of the data flow model to solve the recursive problem, respectively.



**Figure 1.30:** Data flow model for recursive execution

FOR ONLINE USE ONLY  
DO NOT DUPLICATE



**Figure 1.31:** Example of a recursive data flow model during execution

Examples of a recursive algorithm are the sum of n-integers:

**Example 1:** Write a recursive algorithm to find the sum of n-integers.

General algorithm for the recursive algorithm will be given as:

- Step 1: Start
- Step 2: Assign sum = 0 and count = 0
- Step 3: Enter the range, n
- Step 4: Compute count = count + n(count)
- Step 5: Compute sum = sum + i
- Step 6: Repeat step 5 and 4 until count = n
- Step 7: Print sum
- Step 8: Stop

**Example 2:** Implement a recursive algorithm to find the sum of n-consecutive integers in C++.

### Syntax

```
#include<iostream>
using namespace std;
int sum(int n)
{
if (n!=0)
{
return n + sum(n-1);
}
return 0;
}
int main ()
{
int n;
cout<<"Enter the
number:";
cin>>n;
cout<<"sum is:" <<sum(n);
return 0;
}
```

Output 1

The program asks the user to input any number to proceed, as shown in Figure 1.32, so enter any number and press enter.

Enter the number:

**Figure 1.32:** Output 1

Let us input 10 in a prompted message as in Figure 1.33;

Output 2

Enter the number:10

**Figure 1.33:** Output 2:

Now press Enter and the final output will be as in Figure 1.34.

Output 3

Enter the number:10
sum is:55

**Figure 1.34:** Output for a recursive algorithm to find the sum of n-consecutive integers in C++.

### Advantages of a recursive algorithm

The following are the advantages of a recursive algorithm:

- It reduces the number of lines of code in a program.
- It is clear and straight for the programmer to understand.

- (c) Recursive algorithms simply solve complex problems.

### Disadvantages of a recursive algorithm

The following are the disadvantages of a recursive algorithm:

- A recursive algorithm must include condition statements to terminate the recursion, otherwise, the recursion will continue.
- Since the recursive algorithm is repetitive, it requires enough processor time to complete the task.
- It simplifies a solution to the problem when using a recursive algorithm.



### Activity 1.3: Demonstration of the Fibonacci number using C++

- Write the first ten Fibonacci numbers.
- How are numbers related in a Fibonacci series?
- How does the Fibonacci number define a recursive algorithm?
- Write a recursive algorithm to generate the Fibonacci series.
- From your algorithm, state the base case and recursive step.
- Develop a C++ program to display Fibonacci numbers. A program will prompt a user to enter the range of the Fibonacci series and then display the output.

### Exercise: 1.4

- Differentiate between algorithm and program.
- You want to do the homework, and you need a pen. Which algorithm will you use to get a pen from a shopkeeper?
- What is a recursive algorithm?
  - Give two examples of a recursive algorithm.
- How does the base case differ from the recursive step?
  - Which effect will be arisen when the base case is not defined in a recursive function?

### Iterative algorithm

When a recursive algorithm produces many repetitions, it is the best time to use an iterative algorithm. An iterative algorithm involves the repetition of some steps during problem-solving. The iterative algorithm breaks a problem into multiple iterative by adding some features to get the correct solution. It consists of iterative control structures such as *For loop*, *While loop*, and *Do....while loop*. Any iterative event should have a condition to accomplish. For example, if you want to score a goal, you will kick the ball to the goalkeeper and repeat it more than once until you score the goal. So, achieving the plan will be your condition to stop kicking the ball.

There are two types of iterations: definite iteration, which provides a restricted number of times when instruction should be repeated, and indefinite iteration, where instruction is repeated within the loop until another event stops.

**Note: CATE**

An example of the iterative algorithm is the conversion of binary to decimal numbers as follows:

- Step 1: Start the program
- Step 2: Read the binary value, n
- Step 3: Initialise dec=0, i=0
- Step 4: Check whether ( $n \geq 0$ )
- Step 5: Calculate  $r = n \% 10$ ,  $dec += r * pow(2, i)$ ;  $n = n / 10$ ,  $i++$
- Step 6: Print the value of the dec
- Step 7: Stop

An example of a C++ program to represent the implementation of an iterative algorithm activity is as follows:

```
#include <iostream>
#include <cmath>
using namespace std;
int main()
{
    int n, dec=0, i=0, r;
    cout << "Enter any binary number: ";
    cin >> n;
    while (n > 0)
    {
        r = n % 10;
        n /= 10;
        dec += r * pow(2, i);
        ++i;
    }
    cout << "Equivalent Decimal number: " << dec << endl;
    return 0;
}
```

The output of the program is as in Figure 1.35.

```
Enter any binary number: 1001
Equivalent Decimal number: 9
```

**Figure 1.35:** Output of the program

- (i) The iterative algorithm is more efficient than the recursive algorithm in terms of speed of execution and memory usage.
- (ii) The iterative algorithm can also write any program that can be written using a recursive algorithm

**Exercise: 1.5**

1. (a) Identify the real-life example where iteration is applied.  
(b) Write an iterative algorithm to express the example.
2. What are the advantages of using iterative over recursive algorithms?
3. (a) Design an algorithm to find the sum of the digits made from a positive number.  
(b) Create the C++ program to implement the algorithm.
4. (a) Change the following recursive algorithm to find the reverse of the number in iterative form  
Step 1: Start  
Step 2: Enter the value of n  
Step 3: Assign  $r = 0$ ,  $sum = 0$   
Step 4: If  $n > 0$

$r = n \bmod 10$   
 $sum = sum * 10 + r$   
 $n = n \bmod 10$

Step 5: Repeat step 4 until the condition fail

Step 6: Print Sum

Step 7: Stop

- (b) Develop a program of the given algorithm to display the output.

## Searching algorithm

### Meaning of search algorithms

You have studied how data is organised in different data structures such as an array, pointer, stack, and linked list. The most important operation conducted in a linked list is a search algorithm. The searching for the item within the list, involves the following:

- Determine if the data is within the list or not.
- Check if the data are arranged for quickly finding the location of data.
- Determine where the deleted data or item can be found.
- To simplify searching for data or items within the list, each data set should have a unique identified key.

For example, if you want to search for details of a particular student from a school database, student ID as a uniquely identified key should be used to search student details. You will compare that key with the targetted records to find the required data.

Generally, the search algorithm is designed to find or retrieve an item from the list of data. There are two basic types of search algorithms, which are:

- Linear search
- Binary search

### Linear search

Suppose you want to find a particular car number from a list of car numbers. The easiest way is to compare your car number (target) with all numbers within the list by starting from the first number to the last number. Searching is over when you reach the point where

the number equals the target. A system of finding data among several elements within the list by comparing each value with the targeted value is known as a linear search. Sometimes the linear search is known as a sequential search.

A linear search can be performed either within a sorted or unsorted list.

The linear search algorithm derived from the given scenario is:

Step 1: start

Step 2: Compare each number with the targeted number

Step 3: If the present number matches the target, declare victory and stop

Step 4: If the present number is not equal to the target, move to the next number and repeat step 2.

Step 5: stop

Example of a C++ program to implement the linear search algorithm activity derived from the given scenario

```
#include<iostream>
using namespace std;
int a[50], item, n, i;
void linearSearch()
{
    for (i=0; i<n; i++)
        if (item == a[i])
            cout<<"the item is found at location "<<+i;
    return ;
}
int main()
{
    cout<<"Enter number of cars within the list: ";
    cin>>n;
    cout<<"Enter all numbers within the list:"<<endl;
    for (i=0; i<n; i++)
    {
        cin>>a[i];
    }
}
```

```

cout<<"Enter number to
search: ";
cin>>item;
linearSearch();
return 0;
}

```

The output of the program will be in Figure 1.36:

```

Enter number of cars within the list: 4
Enter all numbers within the list
56
12
35
78
Enter number to search: 12
the item is found at location 2

```

**Figure 1.36: Output of the program**

### Advantages of the linear search algorithm

- A linear search is performed very fast if a list contains few elements.
- Linear search does not need to be sorted; anyhow data arranged, searching is possible.
- Any operation within a list, including insertion and deletion of the items, will not affect the linear search.

### Disadvantages of the linear search algorithm

- For the long list, linear search is very slow
- Since linear search compares all data within the list, it needs more space and time to complete searching.

Generally, the linear search seems to be quietly inefficient. Suppose you have 1000 items, and the searched item is at the last position. Then, you will pass through 999 items without getting the value. However, the linear search can work on any sorted or unsorted data set.



### Activity 1.4: Searching a word in a paragraph

Write five sentences to explain about central processing unit (CPU). Your explanation should include the components and functions of each element. Then, write a linear search algorithm to find the word ALU.

### Exercise: 1.6

- What is a linear search algorithm?
- What are the significances of using linear search in real life?
- Consider a list of the number with their position as represented in Table 1.2

**Table 1. 2: List of number and their respective position**

Positions in a list	Value
0	3
1	5
2	2
3	9
4	6
5	1
6	8
7	7

- Write an algorithm to search number '2' from the list of the data value.
- Develop a C++ program for the algorithm you have designed in part (i).

## Binary search

You have realised that sequential or linear search is inefficient for an extensive list. Therefore, if you want to search data from the list that contains many data, use binary search. The binary search uses the divide and conquers approach to search data within the list. Binary search divides the list into two portions when the list is sorted from the smallest value to the biggest value. After dividing the list into two halves, compare the target with the middle item, then decide which half to take up or down the middle value. For example, suppose you want to search for a number from the phone book, where names are arranged in ascending order. Instead of looking for a number from the first to the last, you can start from the middle and see whether to go up or down depending on the position of a targeted name. The following is the algorithm and C++ program to explain the Binary search.

### Algorithm

Step 1: Start

Step 2: Sort the list from small to the large value

Step 3: Find the middle value of the list

Step 4: If the middle value of the list and the targeted value are equal, declare the victory and stop

Step 5: If the middle item is less than the target, then remain with only upper items of the middle value, and from the list and repeat step 2 using the new list.

Step 6: If the middle item is greater than the target, then continue with the lower items list of the middle value, repeat step 2 using the new list.

Step 7: stop

Example of a C++ program to show the implementation of a binary search algorithm

```
#include<iostream>
using namespace std;
int a[50], item, n, loc,
beg, mid, end, i;
void binarySearch()
{
    beg = 0;
    end = n - 1;
    mid = (beg+end) / 2;
    while((beg<=end) && (a[mid]
    !=item))
    {
        if (item<a[mid])
            end = mid - 1;
        else
            beg = mid + 1;
        mid = (beg + end) / 2;
    }
    if (item == a[mid])
        cout<<"the item is found
        at location "<<mid +
        1<<endl;
    else
        cout<<"Item does not
        exist"<<endl;
}
int main()
{
    cout<<"Enter number of
    items in the list: ";
    cin>>n;
    cout<<"Enter all numbers
    within the list"<<endl;
    for (i=0; i<n; i++)
    {
        cin>>a[i];
    }
    cout<<"Enter number to
    search: ";
    cin>>item;
    binarySearch();
    return 0;
}
```

The output of the C++ program (see Figure 1.37).

```
Enter number of items in the list: 5
Enter all numbers within the list
56
21
87
91
99
Enter number to search: 100
Item does not exist
```

**Figure 1.37:** Output of the program

### Advantages of binary search

- Binary search narrowed the long-distance searching by dividing the list into two halves.
- When the list of data is extensive, binary search works more efficiently than linear search.

### Disadvantages of binary search

- Program a binary search algorithm is complicated compared to a linear search
- It is time-consuming to solve, especially if the binary search has many items.



### Activity 1.5: Developing a linear and binary search program

#### Resources:

Student score sheet, computer with C++ compiler

#### Procedure:

- Record current marks for computer science subject from all students
- Set 50% to be a pass mark
- Write a linear search algorithm to find the number of students who passed the exam

- Write a binary search algorithm to find the number of students who failed the exam
- Use C++ programming language to develop a linear and binary search program
- Compile and run the program
- Use the program to search the number of students who have average pass marks and above in a class
- Take the snapshot of the output and save it in a word processor file

### Exercise: 1.7

#### Answer the following questions

- Why do we need a search algorithm?
- What are the advantages of using binary search over sequential search algorithms in solving a problem?
- Assume that you are required to develop a program using the search algorithm, binary or linear. Which search is best for you? Give reasons to support your answer.

### Sorting algorithm

Data should be arranged in ascending or descending order due to different reasons. You can organise data for easy searching, reading, retrieving, or otherwise. For example, a class teacher finds each student's position in a class after placing the students' marks in ascending order. The algorithm responsible for putting data within a

list in a required order is known as the sorting algorithm. The sorting algorithm uses a comparison operator or divides and conquer technique to decide the new order of elements within the data structure.

### Types of sorting algorithms

There are various types of sorting algorithms, including:

- Selection-sort
- Insertion-sort
- Bubble-sort

(d) Heapsort

- Quick-sort
- Merge sort

### Selection-sort

Selection-sort involves sorting by selecting elements in the list and shifting them to their required position. The algorithm locates the smallest element at the top of the unsorted portion of the list. The selection sort is most applied in an array. Figure 1.38 shows a selection-sort algorithm.

Unsorted list

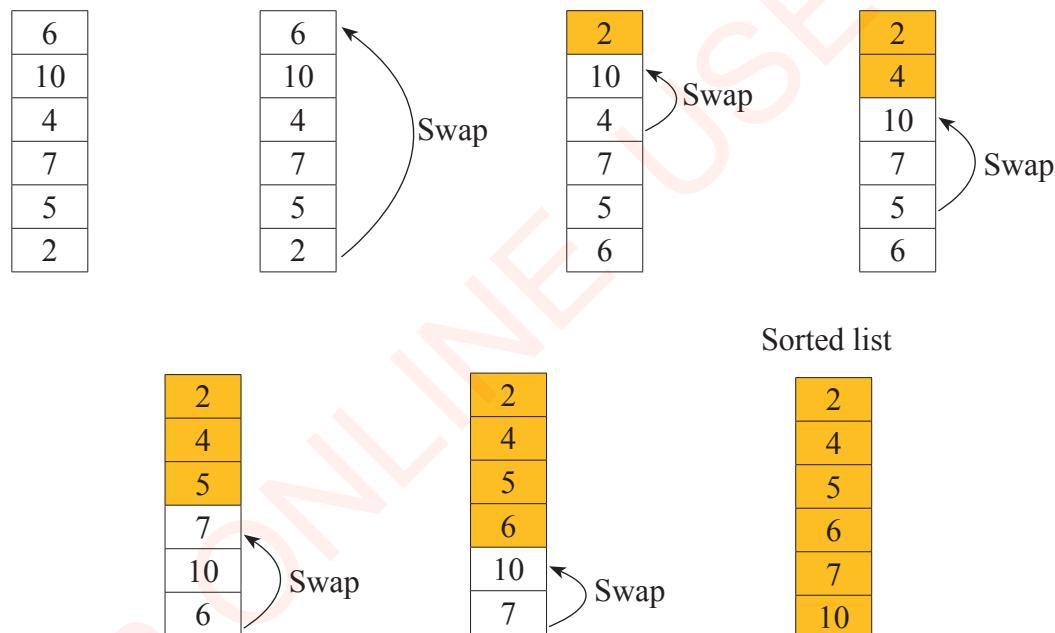


Figure 1.38: Selection sort

Consider the following example for implementing the selection sort algorithm in C++.

```
#include <iostream>
using namespace std;
int findSmallest (int[],int);
int main ()
{
    int school[6] =
{20,8,16,4,1,10};
    int pos,temp,pass=0;
    cout<<"\n Input list of
elements to be Sorted\n";
    for(int i=0;i<6;i++)
    {
        cout<<school[i]<<"\t";
    }
    for(int i=0;i<6;i++)
    {
        pos = findSmallest (school,i);
        temp = school[i];
        school[i]=school[pos];
        school[pos] = temp;
        pass++;
    }
    cout<<"\n Sorted list of
elements is\n";
    for(int i=0;i<6;i++)
    {
        cout<<school[i]<<"\t";
    }
    cout<<"\nNumber of passes
required to sort the
array: "<<pass;
    return 0;
}
```

```
int findSmallest(int
school[],int i)
{
    int ele_small,position,j;
    ele_small = school[i];
    position = i;
    for(j=i+1;j<6;j++)
    {
        if(school[j]<ele_small)
        {
            ele_small = school[j];
            position=j;
        }
    }
    return position;
}
```

### Insertion-sort

Insertion-sort puts an element in its proper place by comparing the value of data and considering the position of a particular data. Insertion-sort divides a list into sorted and unsorted sublists. Then, elements from the unsorted sublist are placed in the correct position in the sorted sublist. Insertion sort can be applied to sort data in linked data, as seen in Figure 1.39.

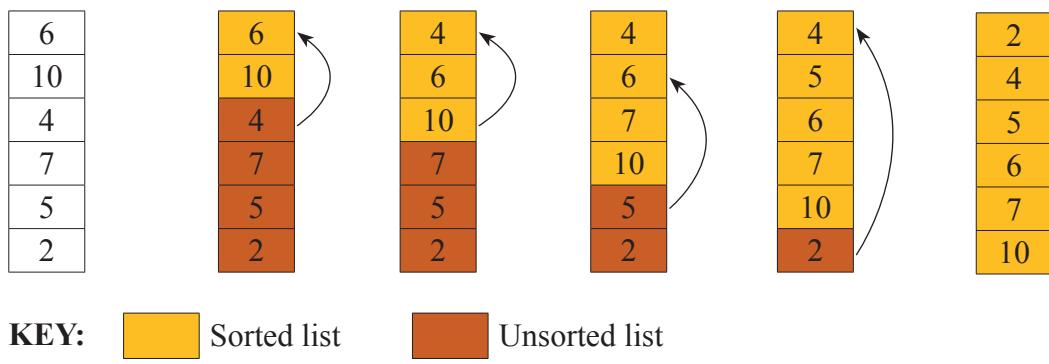


Figure 1.39: Insertion sort

The general algorithm for writing insertion sort will be:

**Step 1:** Start

**Step 2:** set temp = y [z]

**Step 3:** set i = z - 1

**Step 4:** Repeat while temp <= y [i]

    set y [i + 1] = y[i]

    set i = i - 1

        [end of inner loop]

**Step 5:** Set y [i + 1] = temp

        [end of loop]

**Step 6:** Exit

Insertion sort implementation using C++ language

Example of a program using C++ language which to sort the marks of 12 students is as follows.

```
#include<iostream>
using namespace std;
int main ()
{
    int studentMarks[12] = { 20,
        45, 28, 24, 91, 55, 73, 80,
        58, 99, 49, 45 };
    cout<<"\nInput list is \n";
    for(int i=0;i<12;i++)
    {
        cout<<studentMarks[i]<<"\t";
    }
    for(int k=1; k<12; k++)
    {
        int temp = studentMarks[k];
        int j= k-1;
        while(j>=0 && temp <=
        studentMarks[j])
        {
            studentMarks[j+1] =
            studentMarks[j];
            j = j-1;
        }
        studentMarks[j+1] = temp;
    }
    cout<<"\nSorted list is \n";
    for(int i=0;i<12;i++)
    {
        cout<<studentMarks[i]<<"\t";
    }
}
```

## FOR ONLINE USE ONLY

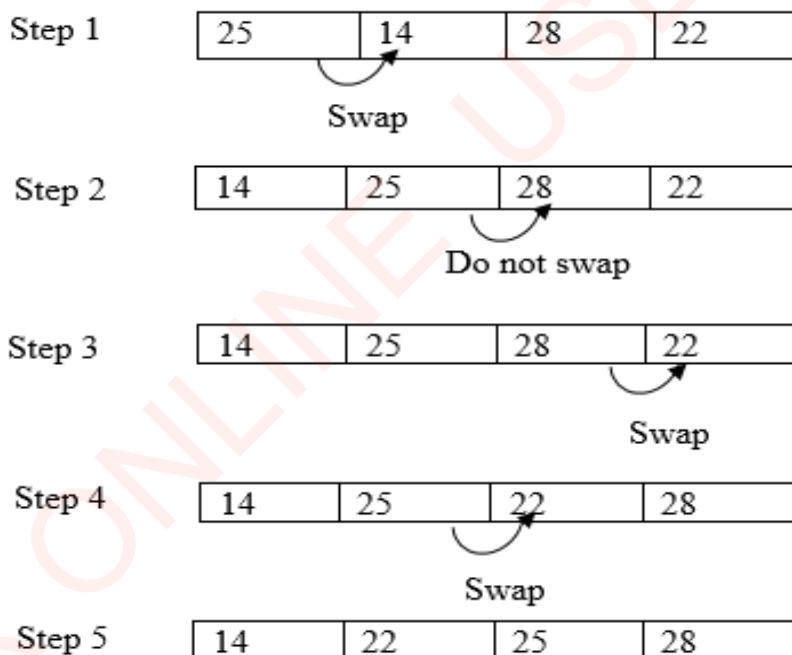
The output of the program is shown in Figure 1.40.

```
Input list is
20      45      28      24      91      55      73      80      58      99      49      45
Sorted list is
20      24      28      45      45      49      55      58      73      80      91      99
```

**Figure 1.40:** Output for insertion sort implementation using C++

### Bubble sort

Bubble sort is the comparison-based type of sorting algorithm. The bubble sort algorithm sorts data by comparing and swapping elements if they are not in the required order. It continues with the swapping of data until all data is sorted. For the data which is arranged horizontally, such data in an array, bubble sort starts by comparing the first two elements and then swapping them if they are not arranged correctly, then arranging the second and third elements. Bubble sort is a stable sorting algorithm but inefficient for large data volumes. Figure 1.41 shows an example of the bubble sort algorithm.



**Figure 1.41:** Bubble sort algorithm

Implementation of bubble sort:

The following is algorithms and C++ program to implement the bubble sort algorithm

### Algorithm

Step 1: Start

Step 2: Compare the first and second element

Step 3: If the first element is greater than the second element of an array, swap them

Step 4: If the first element is less than the second element of an array, move to the next elements

Step 5: Repeat step 1 until all data is sorted within the list

Example of a C++ program for bubble sort algorithm implementation:

```
#include<iostream>
using namespace std;
int main()
{
    int n, i, arr[50], j, temp;
    cout<<"Enter the size of array: ";
    cin>>n;
    cout<<"Enter "<<n<<" Numbers"<<endl;
    for(i=0; i<n; i++)
        cin>>arr[i];
    for(i=0; i<(n-1); i++)
    {
        for(j=0; j<(n-i-1); j++)
        {
            if(arr[j]>arr[j+1])
            {
                temp = arr[j];
                arr[j] = arr[j+1];
                arr[j+1] = temp;
            }
        }
    }
    cout<<"\nThe sorted array is: \n";
    for(i=0; i<n; i++)
        cout<<arr[i]<<" ";
    cout<<endl;
    return 0;
}
```

## FOR ONLINE USE ONLY

The output of the C++ program to implement bubble sort is as shown Figure 1.42.

```
Enter the size of array: 5
Enter 5 Numbers
```

```
45
26
89
12
50
```

```
The sorted array is:
12 26 45 50 89
```

**Figure 1.42:** Output of the program

**Example 1:** Sort (5, 1, 12, 6, 23) using the bubble sort algorithm

**Solution:**

Step 1: (5, 1, 12, 6, 23) → (1, 5, 12, 6, 23)

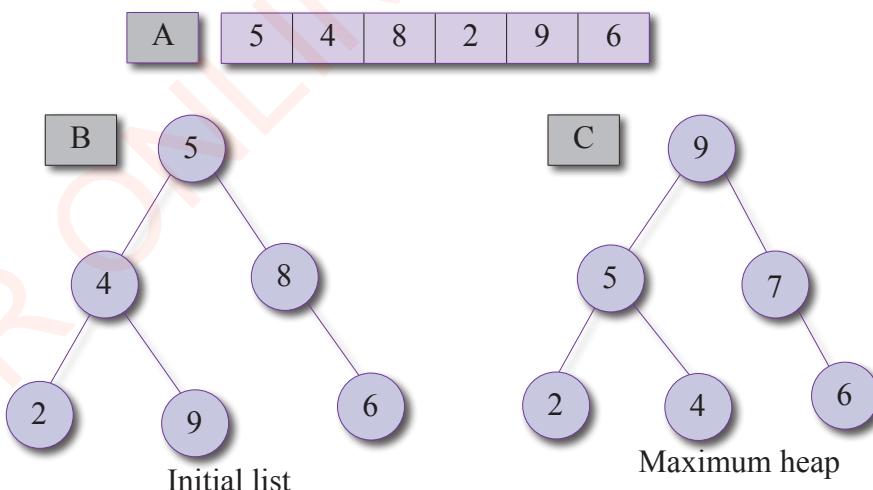
Step 2: (1, 5, 12, 6, 23) → (1, 5, 12, 6, 23)

Step 3: (1, 5, 12, 6, 23) → (1, 5, 6, 12, 23)

Hence, all elements are sorted.

### Heapsort

Heapsort is a sorting method based on comparing elements. It is like a selection sort where you first find the smallest element in the list and position it at the beginning of the list. This process should be repeated to all elements within the list. Unlike selection sort, heapsort does not use much time with a horizontal arrangement of the unsorted element; instead, it has a way of quickly sorting the largest elements. Figure 1.43 shows an example of the quicksort algorithm.



**Figure 1.43:** Heapsort

Using the diagrams in Figure 1.43, you can now generalise writing an algorithm for the heap-sort technique as follows:

**Step 1:** Start

**Step 2:** From the given data, build a max heap from the given data such that the root is the highest element of the heap.

**Step 3:** Remove the root, i.e. the highest element from the heap, and swap it with the last element of the heap.

**Step 4:** Then, adjust the max heap not to violate the max heap properties (heapify).

**Step 5:** The above step reduces the heap size by 1.

**Step 6:** Repeat the above three steps until the heap size is reduced to 1.

**Step 7:** Stop.

Remember, to sort the given dataset in increasing order, you first construct a maximum heap for the given data as shown in the previous heap-sort algorithm.

Example: The following code demonstrates the implementation of heap-sort using C++.

```
#include <iostream>
using namespace std;
void Load(int arr[ ], int n, int root)
{
    int largest = root;
    int l = 2*root + 1;
    int r = 2*root + 2;
    if (l < n && arr[l] > arr[largest])
        largest = l;
    if (r < n && arr[r] > arr[largest])
        largest = r;
    if (largest != root)
    {
        swap(arr[root], arr[largest]);
        Load(arr, n, largest);
    }
}
void loadSort(int arr[ ], int n)
{
    for (int i = n / 2 - 1; i >= 0; i--)
        Load(arr, n, i);
    for (int i=n-1; i>=0; i--)
    {
        swap(arr[0], arr[i]);
        Load(arr, i, 0);
    }
}
void displayArray(int arr[ ], int n)
{
    for (int i=0; i<n; ++i)
        cout<<arr[i]<< " ";
    cout<< "\n";
}
int main ()
{
    int heap_arr[ ] = {11,9,6,13,2,8};
    int n = sizeof (heap_arr)/sizeof (heap_
    arr[0]);
    cout<<"Inputed array"<<endl;
    displayArray (heap_arr,n);
    loadSort (heap_arr, n);
    cout<< "Sorted array"<<endl;
    displayArray(heap_arr, n);
}
```

## FOR ONLINE USE ONLY

TOPIC 1: DATA STRUCTURE

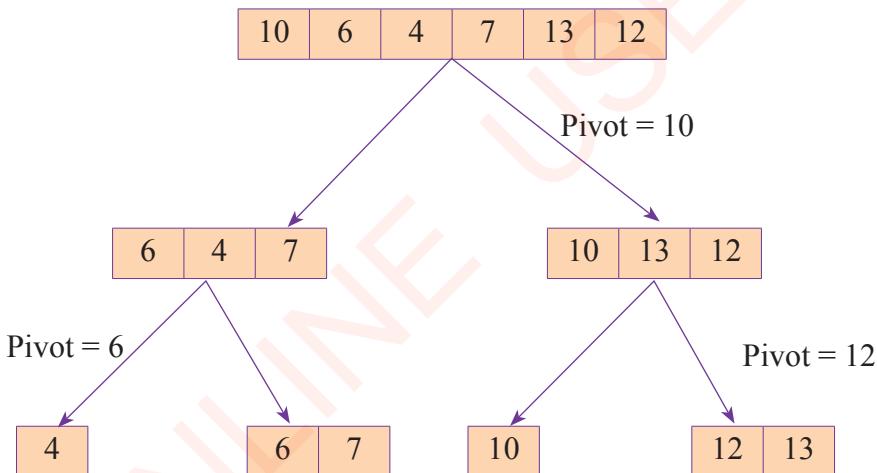
The output of the heap-sort program is as shown in Figure 1.44:

```
Inputed array
11 9 6 13 2 8
Sorted array
2 6 8 9 11 13
```

**Figure 1.44:** The output of the heap-sort program

## Quicksort

Quicksort uses the divide and conquer sorting technique. The list is portioned into two sublists, and each will be sorted and joined to form one list again. It uses the recursion method to implement sorting. Quicksort divides a list into lower and upper sublists separated by a pivot. A pivot is a selected element from the list that is more significant than all elements in a lower sublist and less significant in an upper sublist. Quicksort can be applied to sort data within an array. Figure 1.45 shows an example of the quicksort algorithm.



**Figure 1.45:** Quick sort

## MergesSort

Merge sort is also a divide and conquer sorting algorithm. It divides the list into two halves and then merges the two sorted lists. Merge sort starts by dividing the list into sublists and continues dividing sublists into two more halves while swapping the elements, then combining swapped elements to form a completely sorted list. Figure 1.46 shows an example of the quicksort algorithm.

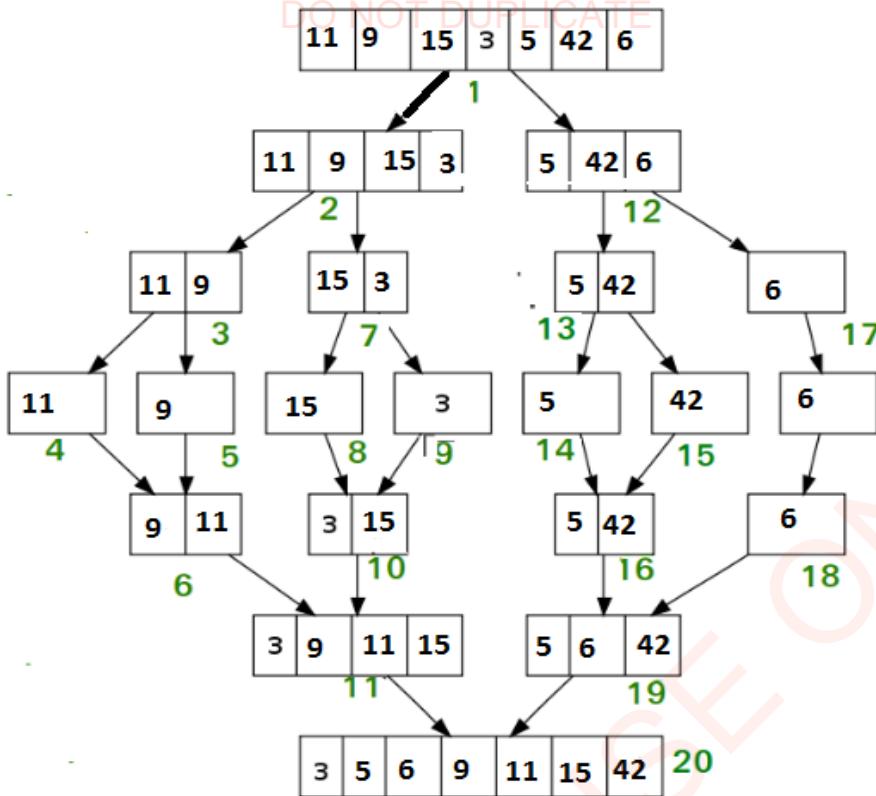


Figure 1.46: Merge sort

The following are things to consider to choosing the sorting algorithms for the application:

### Stability of the sorting algorithms

The stability of a sorting algorithm is considered by how it treats identical elements. Stable sorting algorithms maintain the order of equal or repeated elements, which is impossible for unstable sorting. This is to say, stable sorting preserves the order of two similar elements within the list. Suppose you want to sort the students according to their grades after already sorted based on their names; you must use a stable sort algorithm such as merge and insertion sort to sort similar grades without affecting their names. However,

it is possible to make unstable sorts like heap and quick sort stable.

### Execution time

This explains a running time algorithm that can be set for data processing. It explains the average performance of the algorithm on every set of data. Usually, execution time depends on recursive calls, memory usage, and comparisons. Quicksort and merge sort are examples of fast sorting algorithms.

### Memory usage

Memory usage depends on the size of the input data to be sorted. An example of a sorting algorithm that does not need extra memory is the insertion and bubble sort. Otherwise, merge sort and quick

sort requires more memory to complete sorting.

**Note:** Depending on some factors, every sorting algorithm can be the best or worst. These factors are based on the problem to be solved and the data to be manipulated.

### Exercise: 1.8

Answer the following questions:

1. Use the bubble sort algorithm to arrange the following numbers in ascending order: 9, 3, 10, 2, 8, 5, 1.
2. Write a C++ program to display the sorted list in question (1).
3. Given the list of the elements: 6, 10, 25, 8, 45, 15, whereby the first three elements are in order, how many key comparisons are executed to move 8 to its proper position using bubble sort?
4. Given the list of elements to be sorted: 25, 15, 22, 11, 24, 32, 13, 70, 33, 55, 14, use bubble sort to determine the resulting list after six steps.

### Chapter summary

1. The data structure is the standard way of storing and organising data in a computer for quick and efficient searching, sorting, and accessing data.
2. The data structure available in the computer and generally defined within a compiler is known as a primitive data structure. The primitive data structure stores only one type of data. Examples of primitive

data structures are String, Boolean, character and integer. A user-derived data structure that can store different data types is the non-primitive data structure. Examples of non-primitive data structures include arrays, linked lists and stacks.

3. The data structure in which the number of elements is fixed is the static data structure. An example of a static data structure is an array. The data structure in which the number of elements is not fixed is known as a dynamic data structure. An example of a dynamic data structure is a linked list.
4. An array is a static data structure. It contains identical types of data. The elements in the array are stored in consecutive memory locations. An example of an array is  $a[3] = \{7, 9, 11\}$ .
5. Stack is a non-primitive linear data structure in which data insertion and deletion occur from only one end, known as the *top*.
6. Pointers are usually auxiliary variables that indirectly allow users to access other variables' values. For example, int x, \*p; "x" is the numerical variable in the declaration while "p" is the pointer to the number.
7. Records are the type of data structure comprising a series of related data. Unlike an array data structure which includes data of the same type, a record contains data for different kinds.
8. The queue is a non-primitive linear data structure in which data insertion and deletion occur from two opposite ends, rear and front.

9. A linked list is a dynamic data structure where nodes are connected using a pointer. Each node in a linked list contains data and addresses related to the next node in a list.
10. User-defined data structures are data structures implemented independently within a computer program. These data structures are built by combining primary and built-in data types and then associating operations. Examples of such data structures are Class, Structure, Union, Enumeration, and Typedef defined Datatype
11. A binary tree is a non-linear dynamic data structure in which two or more children make each node. Binary trees are used to implement binary search trees and binary heaps for efficient searching and sorting.
12. The algorithm is the correct sequence of steps of instructions required to solve a problem.
13. The algorithm should be precise, finite, definite, effective, etc.
14. The approach to designing an algorithm can either be recursive or iterative.
15. The recursive algorithm explains the algorithm approach where the solution to the main problem depends on solutions of more minor instances of the same problem.
16. An iterative algorithm involves the repetition of some steps during problem-solving
17. Searching algorithms are designed to find or retrieve an element stored in a data structure. Based on the type of search operation, these **DUP** algorithms are classified into a linear and binary search
18. Linear search is the algorithm that traverses the list entirely and matches each list's value with the targeted value.
19. The binary search uses the divide and conquers technique to find elements after sorting the list.
20. The sorting algorithm is an algorithm that puts elements of a list into ascending or descending order.
21. Types of sorting algorithms include selection sort, merge sort, heap sort, insertion sort, bubble sort and quick-sort
22. Bubble sort is the sorting algorithm that works by repeatedly swapping the adjacent elements if they are in the wrong order.

### Revision Exercise

1. (a) What is meant by the term data structure?  
 (b) Give out the characteristics of static data structure.
2. (a) What is an array?  
 (b) Differentiate the array from the linked list.  
 (c) Explain the advantages and disadvantages of an array over the linked list.  
 (d) Describe a programming situation that will lead a programmer to choose an array or a linked list.
3. (a) What is the difference between array and pointer data structure?

- (b) With examples, briefly describe the following as used in the data structure:
- Stack
  - Queue
4. Describe the stack implementation in the C++ program in an array.
5. By using illustrations, describe operations done in the stack data structure.
6. Describe four operations that the queue can perform.
7. Sort the numbers: 55, 44, 83, 26, 47, 66, 92 and 72 in a binary tree structure and arrange them in Pre-Order, In-order, and Postorder.
8. State three (3) features of a suitable algorithm.
9. Design an algorithm for a program that inputs the radius of a circle from a user. The user will be asked by the program to calculate diameter, circumference, or area. The result will be calculated and then displayed.
10. Write an algorithm to find squares of the number.
11. Write a C++ program for the following algorithm
- Step 1: Start  
 Step 2: Enter two numbers A and B  
 Step 3: Check if A is greater than B, if yes go to step 4 else go to step 5  
 Step 4: Print A is greater than B  
 Step 5: Check if B is greater than A, if yes go to step 6 else go to step 7  
 Step 6: Print B is greater than A  
 Step 7: Print A is equal to B  
 Step 8: Stop
12. (a) Write an algorithm to find the length of a password.  
 (b) Write a C++ program to implement the algorithm made in part (a).
13. (a) What is linear search?  
 (b) Explain steps to implement linear search.
14. (a) What is the binary search?  
 (b) The following form an array: 10, 31, 0, 9, 38, 20, 13, 24, 7 and 200. Why is it impossible to locate key "0" using binary search?
15. (a) By showing steps clearly, show how binary search is implemented to find an element in the following Array, and let the searched key be 56.
- | Elements | Index |
|----------|-------|
| 10       | 0     |
| 12       | 1     |
| 24       | 2     |
| 29       | 3     |
| 37       | 4     |
| 40       | 5     |
| 51       | 6     |
| 56       | 7     |
| 68       | 8     |
- (b) Use a C++ program to implement the given algorithm.
16. (a) What is bubble sort?  
 (b) Explain the implementation of bubble sort.

17. Read the following C++ code and answer the questions that follows:

```

int a[5] = {2, 5, 4, 1, 3},
i, k;
int x;
for(i = 0; i<4; i++)
{
    for(k = 1+i; k<5; k++)
    {
        if(a[i]<a[k])
        {
            x=a[i];
            a[i] = a[k];
            a[k] = temp;
        }
    }
}
    
```

- How many elements are included in an array?
- What type of sorting algorithm is implemented on the given code?
- In which order the array “a” will be sorted?
- Which variable will be used temporarily in the given code? Justify your answer.
- Write a complete C++ program to display the sorted array “a.”

### Project to develop a simple program using C++

Ketox is a service rendering application system. The application aims to create a bridge between service providers and clients to ease the accessibility of business-related services. The application has two sides, one for the service provider and another for the clients. Each side defines business privileges to the particular side.

The client-side has the following privileges:

- Search for a service provider and order the service.
- Get all the necessary information about any service provider.
- Pay for any ordered service through any mobile money service application.
- Rate and comment upon the services provider.

The service provider has the following privileges:

- Receive requests from clients.
- Get a short portfolio of all clients who need service.
- Approve payment soon after receiving the message of the money sent by the customer from mobile money application services.
- Rate and comment upon the clients.

Design the system's algorithm, then use it to implement the system using a C++ programming language.

# Information systems

## Introduction

Nowadays, people and organisations use various information systems to handle their daily activities in healthcare, education and training, financial transactions, administration, agriculture, tourism and other sectors. For example, the Tanzania Institute of Education (TIE) online library gives students and teachers textbooks and video conferencing facilities to facilitate business meetings, and instant messages. In this chapter, you will learn about these information systems and how to manage and apply them in real life. Important issues related to information systems in general, like database systems, relational databases, and database management, will be covered. The competency achieved will enable you to design, develop, manage, and use various information systems.

## Overview of Information Systems

Information systems facilitate corporate and individual operations, interaction, and decision-making. An information system is usually a computer-based system designed to support operations, management, and decision-making. Also, Information Systems are combinations of hardware, software, and telecommunications networks that people build and use to collect, create, and distribute valuable data, typically in organisational settings. An information system is made of different components that work together to provide value to an organisation; it includes hardware, software, data, people, and procedure.

## Data and information

Data is a collection of unprocessed raw facts, which have no meaningful context, whereas Information is processed data organised and presented in a meaningful context. Examples of data are numbers, symbols, characters, texts, and multimedia forms, e.g., 125A. Four data types representing these facts are shown in Table 2.1.

Examples of information are such as reports, transactions, and processed data. The primary function of the Information System is to receive the raw data (not processed) and then process it to produce the information with meaningful contexts, such as reports and other information formats.

**Table 2. 1:** Forms of data representation

Data	Represented by
Alphanumeric data	Numbers, letters, and symbols characters
Image data	Graphic images and pictures
Audio data	Sound, noise, or tones
Video data	Moving images or pictures

### Characteristics of information

Information is a collection of data organised and processed to have additional value beyond the value of individual facts. An examination officer, for example, through an information system, may want individual examination results summarised to see the average, maximum and minimum grades achieved. Information originating from information systems can help people in their organisations to perform tasks more efficiently and help decision-makers achieve their organisation's goals. Many businesses and other parties assume that reports are based on correct and quality information. However, this is not always true. Therefore, it is necessary to ensure that an Information System's characteristics regarding the information to be produced are followed. Furthermore, there are several characteristics of valuable information to consider when handling the Information Systems as follows:

#### Accessibility

Information should be easily accessible by authorised users to obtain it in the proper format and at the right time to meet their needs. For example, the NECTA Examination system from which form IV and VI results are easily accessible to all students, parents, schools, and other

stakeholders. On the other hand, other parts of this system are restricted to NECTA staff for administration matters; so, those who are authorised can access that part. Also, a telephone directory or a phone book on your phone or computer should be easy to obtain or access. It is most often the first place when you look for a phone number. However, because no one stores the entire country's phone books, you can call a mobile phone customer care for numbers beyond what you have or utilise the internet to find out, information about a hotel in a different country, say the Democratic Republic of Congo (DRC).

#### Accuracy

The information generated by the information systems must be accurate enough to be used as intended. Obtaining 100 percent precise information is frequently unattainable because it is likely to be too costly to supply on time. The level of precision varies depending on the situation. For example, information may need to be precise at the operational levels on a supermarket checkout receipt. Accuracy is an essential feature of information systems and the data they create. For example, suppose the government information based on the most recent census incorrectly reflects a rise or fall

in births within a region. In that case, the government's plans and budgets in that region will not reflect the reality. This inappropriate reflection may lead to unrealistic planning such as allocated budget for school and hospitals.

### **Completeness**

An information system's output should include all of the user's details. Otherwise, it could not be a reasonable basis for deciding. For example, suppose an organisation is given information on the cost of selling laptops to a school, but not on the cost of service and maintenance. In that case, the total cost of ownership based on the information provided will be significantly underestimated.

### **Economical or cost-effective**

Information should be accessible at a predetermined price, which may change based on the situation. If obtaining information is expensive, an organisation may opt for less expensive information elsewhere. For example, suppose an organisation wishes to do a market research study on a new product, the survey could cost more than the product's expected initial profit. The organisation would most likely choose a less expensive information source in that circumstance.

### **Flexibility**

Information systems should be flexible to be used for a variety of reasons. A procurement officer in a school, for example, might utilise information on how much inventory is on hand for a specific part to close a deal. The headmaster can use the same information to approve a new inventory, and a school accountant may use the same to secure the required amount of money.

### **Appropriateness and relevance**

The information supplied by information systems should be relevant or appropriate for the task at hand. What is important to one person or organisation might not be necessary to another. The user will get annoyed if the material contains unnecessary data for the task. For example, detailed technical data on equipment and production specifications can be relevant to the line supervisor but may be irrelevant to upper management.

### **Objectivity or dependability**

Objectivity deals with the truth of the information that an information system presents and relies on. Only if you are confident of the information's reliability or objectivity can you use it confidently. For example, you may go straight to the library to choose an appropriate book for an essay on any subject if you have reasonable confidence in the accuracy and objectivity of information obtained from a book or a library. This might be due to the reason that books in the library are authored and edited by specialists in the topic, and the authors' names are usually prominently displayed.

### **Secure**

Unauthorized users should not be allowed to access information. For example, in a college information system, examinations and staff salary information are sensitive; hence, they have to be secured against students and other unauthorised persons.

### **Simple**

Information should be straightforward and in a format that permits it to be examined and used without the addition

of excessive data. It is normal, for example, to summarise financial data or examination results and present them in figures, charts, or graphs. Having too much information can lead to information overload, in which a decision-maker or a consumer is unable to decide what is important.

### Timely

For a reason, it is necessary for the information to be delivered on time. Information from an information system that is obtained too late is useless. For example, if you get a brochure from a theatre and see that your favourite band had a show yesterday, the information is too late to be helpful.

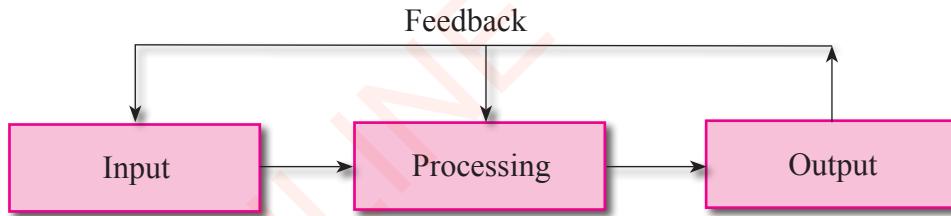
### Verifiable

Information should be able to be verified. This means that you may double-check it for accuracy, perhaps by examining many sources for the same information.

For example, the students' results information in a school or university information system has verifiable original transcripts in physical or online systems that capture and store them.

### Concept of information systems

An information system (IS) is a set of interrelated components that collect, process, store, distribute data and information, and provide a feedback mechanism to meet an objective. Therefore, it involves a cycle that contains (input), manipulates (process), distributes information (output), and offers a corrective reaction (feedback) to fulfil a system goal, as indicated in Figure 2.1. The feedback mechanism is a critical aspect of a system's performance since it aids organisations in achieving their objectives. For example, increasing efficiency and enhancing customer service.

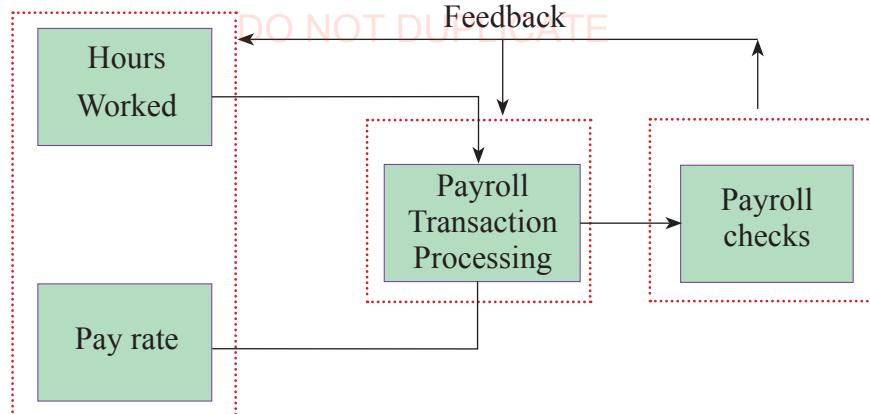


**Figure 2.1:** An information system (input, output, processing, feedback)

### Input

Input is the action of acquiring and capturing raw data in information systems. For example, when creating payroll checks, each employee's number of hours worked must be gathered before the payroll check can be generated and printed. Supervisors must submit their employees worked hours in a payroll system as indicated in Figure 2.2 to generate payroll checks for the given employees.

In the payroll system demonstrated in Figure 2.2, the input goes through a transformation process to produce output with the possibility of improving the input and process through feedback.

**Figure 2.2:** An example of a payroll system

## Processing

The term “processing” in the context of information systems refers to the transformation of data into meaningful outputs. Calculations, data comparisons, alternative actions, and data storage for future use is all part of the process. In a computer system, converting data into valuable information is crucial. Processing can be done manually or with the help of a computer. For example, in a payroll system indicated in Figure 2.2, the number of hours each person worked must be converted to pay per day, week, or month, based on the agreed-upon payment arrangement. Other inputs, in this case, can be an employee ID number and department. To calculate gross compensation, start by multiplying the number of hours worked by the employee’s hourly pay rate. Overtime pay may be included if weekly hours worked exceed 40. Then, to get net pay, deductions such as government taxes, insurance contributions, and loan payments are removed from gross pay. These calculations and comparisons are usually saved in storage devices when they are completed. The term “storage” refers to the process of keeping data and

information, including output, available for future use.

## Output

Producing an output in an information system entails creating valuable information, which is typically in the form of documents and reports. In Figure 2.2 of a payroll system, an output could be the payroll checks for employees, a report on payment for the management, the involved bank, and the employees’ social security fund organisation. In some circumstances, one system’s output can be used as input for other systems. For instance, the output can be used as input for the said/involved social security fund organisations.

## Feedback

Feedback is information from a system used to modify input or processing in the information systems. Errors may trigger the correction of input data or the modification of a process. Consider the case of the payroll system indicated in Figure 2.2. Perhaps an employee’s number of hours worked was 300 rather than 30. Fortunately, most information systems validate data, ensuring that data fall within predefined limits. Because

## FOR ONLINE USE ONLY

an employee will rarely work more than 100 hours in a week, the range for the number of hours worked may be 0 to 100. The information system would identify that 300 hours are outside the acceptable range and notify the user. The feedback is used to double-check and correct the input of 30 hours worked. This mistake would result in unnecessarily high net pay if it went undiscovered.

Figure 2.3 illustrates the components of the information systems in the way a computer receives data (input), manipulates it (process), and provides the results (output). Some output may be different than you imagined when you look at the results offering you some ideas for how to make it better (feedback).

Examples of Information Systems:

1. **Food Information System**
  - (i) **Inputs:** Consumer orders
  - (ii) **Processes:** Processing software
  - (iii) **Output:** Receipts, cook's order list
  - (iv) **Feedback:** Invalid entry message
2. **University /College information system**
  - (i) **Inputs:** Students, faculty, textbooks
  - (ii) **Processes:** Education/Courses
  - (iii) **Output:** Graduates
  - (iv) **Feedback:** Surveys, grades
3. **Video Library Information System**
  - (i) **Inputs:** Rentals, returns
  - (ii) **Processes:** Processing software

(iii) **Output:** Reports, rentals agreement

(iv) **Feedback:** Error reports

#### 4. **In the Car Plant**

- (i) **Inputs:** Raw materials components
- (ii) **Processes:** Assembly line
- (iii) **Output:** Mini-vans
- (iv) **Feedback:** Customer surveys, quality reports

### **Characteristics of information system**

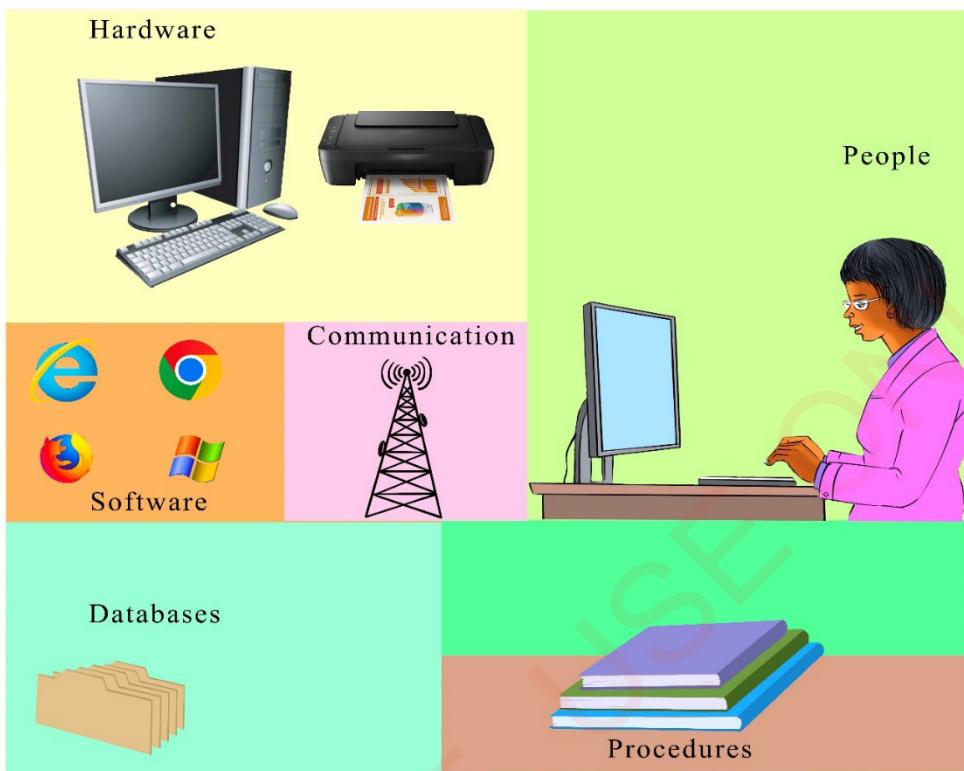
The following are characteristics of the information system:

- (a) Organises data into information
- (b) Analyses information to give knowledge during data processing from input to output.
- (c) Processes data and transform them into valuable information for decision-making.
- (d) Promotes evidence-based decision-making like, producing reports.
- (e) Automation of workflows through computerised processes involving users in each work field.
- (f) Enhances usability by providing an accessible user interface that makes it easy to use.

### **Components of information system**

The computer based information system components comprise of hardware, software, telecommunications (communication channels including computer networks and the Internet), people, and procedures. Information system components are put in place by an organisation to help with its processes

or human activities. People or machines can communicate and work together to achieve a goal in this digital age. This set forms the components of an information system, as indicated in Figure 2.3.



**Figure 2.3: Components of an information system**

These components are interconnected and work together to provide a solution to problems faced by society, allowing them to achieve their objectives. The following is a list of each component's description:

### Hardware

Hardware is any physical/tangible part of the computer. It is utilised to conduct computer input, processing, storing, and output operations. Input devices capture instructional data or instructions that can be transformed into alphanumeric, picture, sound, or video data. Keyboards, mouse, fingerprint and other pointing devices, automatic scanning devices,

and magnetic ink-reading equipment are examples of input devices. Smaller, faster, and more transportable input devices that automate this process are becoming popular in the computer industry. Computer chips with a central processing unit and primary memory are examples of processing devices.

Many automated operations that we undertake today in our offices, banks, colleges, schools, organisations, and as individuals are done swiftly and efficiently with handheld, lower, and medium-range computer equipment. Furthermore, because of powerful supercomputers, complex tasks such as

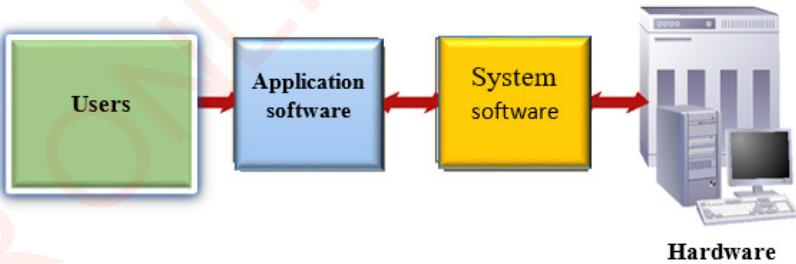
the design of new autos that used to take two to three years are now doable in a relatively shorter period. Output devices handle the result of the computer's processed input. They include printers and computer screens. For example, computers with sensitive touch-screens that may execute certain task or run applications, including opening the Internet or launching a new computer game or word processor.

There have also been numerous special-purpose hardware devices built, such as Event Data Recorders (EDRs). These devices are currently being installed in many places such as, vehicles where they serve as the blackboxes to capture information on driver performance, potential engine faults, and road speed of the vehicle. Many individuals prefer the mobility, functionality, and cost of tablets and smartphones, which give computing power, access to applications, and communications services whenever and wherever the user wants them.

### Software

The software consists of the computer programs that govern the operation

of the computer. There are two types of software; systems software and application software, both of which interact in use, including hardware, as illustrated in Figure 2.4. System software coordinates hardware and software processes. They include Operating Systems like Microsoft Windows and Android, utility programs like Norton antivirus software, and development programs like C++. Applications software includes general application software like Microsoft Office that allows users to perform cross-cutting tasks such as editing text documents, analysing data, and creating graphs. Specific application software solves particular problems such as human resource systems and student systems. All forms of computers, from small handheld computers to massive supercomputers, require both systems and application software. For instance, Google's Android operating system and associated apps like Google Maps are designed specifically for touch-screen mobile devices such as smartphones and tablets.



**Figure 2.4:** Use of application software and systems software and how they relate

### Database as an application software

A database is an organised collection of facts and information, usually two or more related data files. Clients, employees, students, inventory, sales, online purchases, and much more can all be found in an organisation's database. For example, the Online

Loan Application and Management System (OLAMS) by Tanzania Higher Education Students' Loans Board (HESLB) handles over 100,000 applicants for loans for higher education. Data are collected from all interested beneficiaries, mostly from higher learning institutions and intermediate college graduates. Data is collected from people seeking for loans as well as those who have already graduated and are now required to repay. Thousands of them and a Higher Education Students Loans Board (HESLB) staff utilise the system every year. This includes registration, application, follow-ups, and issuing and refunding the loan by beneficiaries and employees, resulting in an increase in digital data and significant database and device storage requirements. The OLAMS systems database makes it simple and convenient to obtain the data, for use in operations and decision-making. For example, HESLB staff can obtain a detailed report of beneficiaries who received loans to study at a specific university.

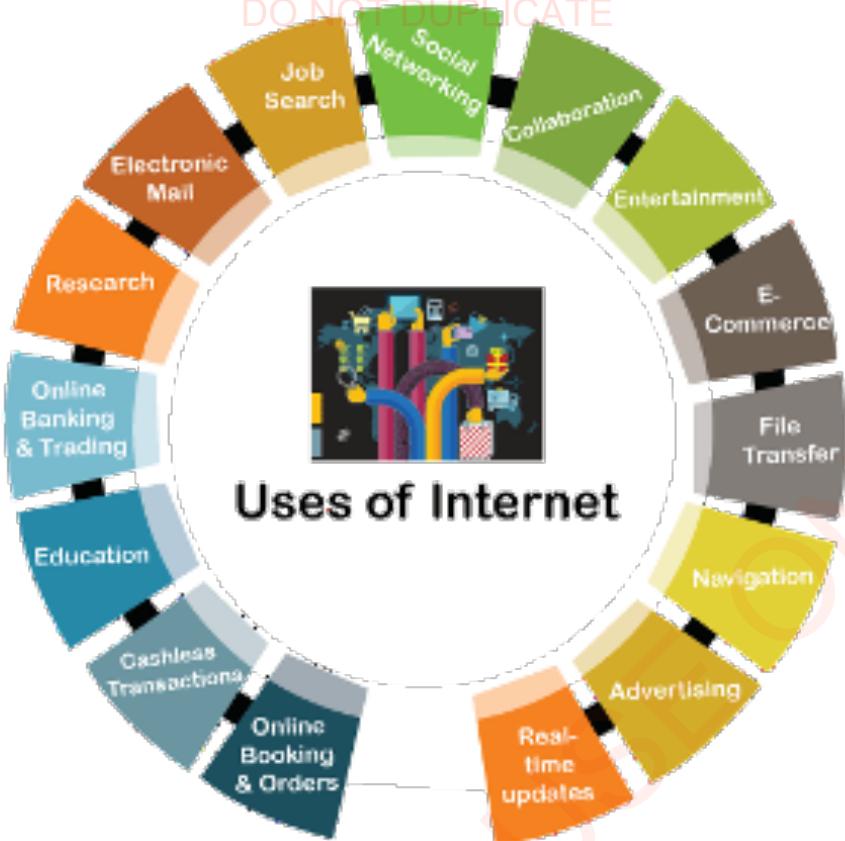
### Telecommunication, Network, and the Internet

Telecommunication is the transmission of information by various technologies over the wire, radio, optical, or other electromagnetic systems. Categorically, telecommunications can take place through wired, wireless, and satellite transmissions. People may now see the news live as it happens, including breaking news events, due to satellite communication networks that have eliminated delays in news transmission. People and organisations of all sizes are using telecommunications worldwide.

Mobile devices such as cell phones and tablets can connect via wireless transmission. People can work from home or on the road because of telecoms. Telecommunications also allow virtual teams of people working on a project to meet and interact without physical meetings. For example, during the COVID-19 pandemic, when the social distance was required, many people were able to work and learn from their homes.

Networks, on the other hand, enable electronic communication by connecting computers and equipment within a building (Local Area Network-LAN) such as an administration block in a school; across the buildings (Metropolitan Area Network-MAN) such as various colleges' buildings in a University; and across the country or countries (Wide Area Network-WAN) such as in the Southern African Development Community (SADC)-wide based company.

Furthermore, the Internet is a vast network that connects computers worldwide. In today's technological era, most organisations do their operations over the Internet. Organisations and individuals make their daily tasks more productive and comfortable through various Internet use. Major uses of the Internet that play a vital role in daily life include online bookings and orders, cashless transactions, education, online banking, and trading. Others include research, electronic mail, job search, social networking, collaboration, entertainment, e-commerce, file transfer, navigation, advertising, and real-time updates, as indicated in Figure 2.5.



**Figure 2.5:** The uses of the Internet that play a vital role in daily life

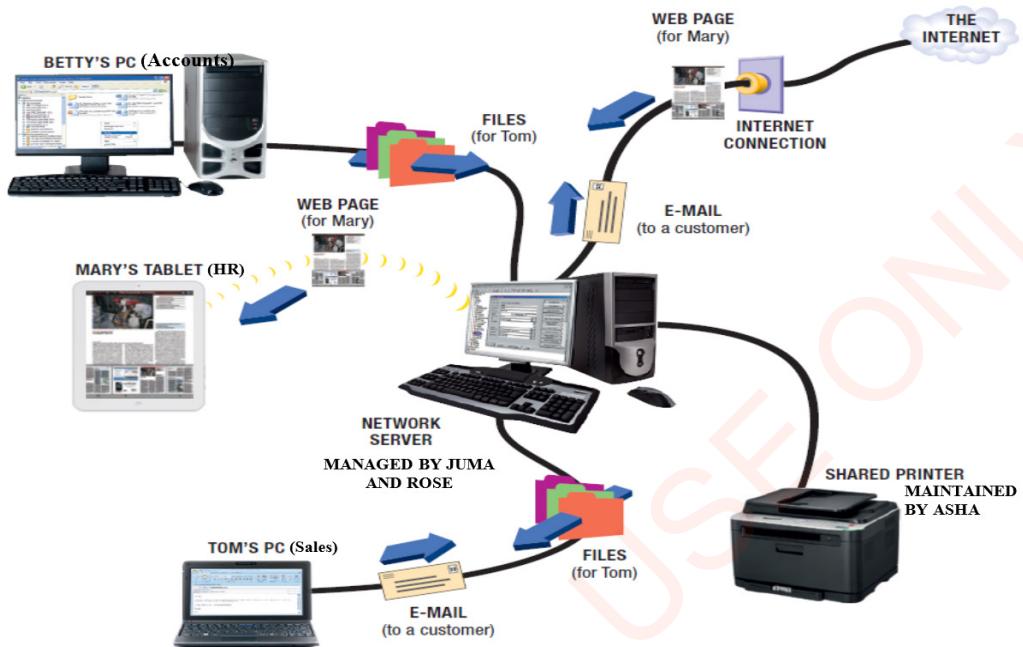
The technology used to create the Internet is also being applied within organisations to develop intranets, which allow people in an organisation to exchange information and work on projects. Through the internet, employees can quickly access the internet; employees can promptly access work resources, collaboration, and fun. An extranet is a network based on web technologies that allow selected outsiders, such as business partners and customers, to access authorised resources of an organisation's intranet. Many people use extranets daily without realising it to track shipped goods, order products from their suppliers, or access customer

assistance from other organisations. Dalsey Hillblom Lynn (DHL) is a large company that empowers customers to serve themselves at their convenience through a corporate extranet. Through its customers' information system, customers can access the DHL extranet to obtain a full range of shipping, billing, and tracking services.

### People

People are the most vital component of computer-based information systems. They make it possible for the computer system to work effectively, improve and solve society's challenges and problems. All persons who utilise, manage, run,

develop, program, and maintain information systems; including IT professionals who handle technical aspects and end-users who work directly with information systems to get outcomes, are considered information systems personnel, as shown in Figure 2.6. End-users include finance executives, marketing representatives, sales persons and manufacturing operators, while IT professionals includes systems analysts, persons, network engineers, and systems administrators.



**Figure 2.6: People - End-users & IT professionals that use, manage, develop and maintain the system**

## Procedures

Procedures are strategies, policies, and guidelines for using a computer-based information system (CBIS) more effectively to meet the organization's objectives. They define the processes to achieve a given result, such as submitting a customer order, paying a supplier invoice, or getting a current business-oriented report at the entry-level. From the end users' and IT professionals' perspectives, reasonable procedures outline how to reach the desired result, who does what and when, and what to do if something goes wrong. People who are adequately taught and follow effective procedures available in various ways can complete tasks faster, save money, better use human resources, and help people adapt to change. Procedures clearly documented, such as in user manuals, systems manuals, and Frequently Asked Questions (FAQs), can drastically minimise training costs and time to mastery. This also applies to the availability of high-level policies and strategies that are aligned with the objectives of these information systems in the organisation.



### Activity 2.1

An information system consists of input, processing/procedure, output, and feedback. Identify an information system in your school or community as a whole and describe how it works based on these elements, and its impact on society. Also, identify the input, processing, and output of the specified information system.



### Activity 2.2

Use an internet search engine or any other means to find an information system example in real life, and describe how its components interlink and work together to achieve a specified goal successfully.

### Exercise 2.1

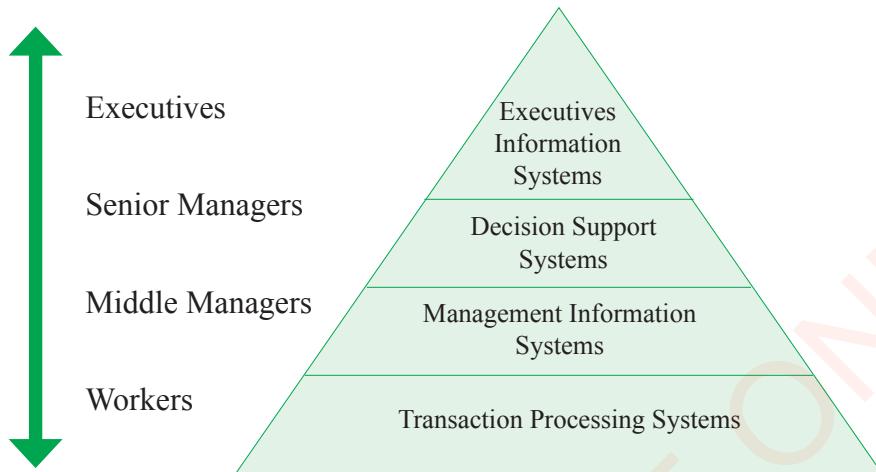
- Using any two examples, describe the importance of information systems in today's digital society.
- Briefly describe two of the valuable characteristics of information that modern information system require. What is the difference between them and their impact when developing, implementing, and using an information system?
- Describe five characteristics of an information system.

- What is a computer-based information system? How does it differ from a traditional file-based system?
- Two of the system components of a computer-based information system are hardware and software. Describe the importance of each and, with an example, explain how they depend on each other.
- Suppose you had to explain the concept of an information system to a family member. How will it be defined? In one paragraph, write a description that best describes an information system to your family using your own words.

### Types of information systems

Today's information systems come in various shapes, sizes, and functions and cover a variety of scopes. While some information systems are unique and others vary somewhat from one organisation to another, they can usually be grouped by their basic functions into four main types of information systems. These types are Transaction Processing Systems (TPS), Management Information Systems (MIS), Decision Support Systems (DSS), and Executive Support Systems (ESS) or Executive Information Systems (EIS). They are also summarised in Figure 2.7 and discussed next, including who uses each system, the output, and trust for each type. Their true potential is to help

employees and managers to provide all necessary information in the organisation's operations, control and assessment, and decision-making. These systems can cut costs, uncover new opportunities, increase profits, or improve effectiveness.



**Figure 2.7: Four main types of information systems**

### Transaction processing systems

An information system that captures and processes data from day-to-day business activities is known as a Transaction Processing System (TPS). It ensures that a business transaction is completed successfully and keeps track of transactions using a database. Effective TPS can speed up business activities, increase reliability, and reduce clerical costs. For example, buyers who buy a book from a store may pay with a credit card. A transaction processing system takes a customer's credit card number, connects with their bank, and authorises or refuses the purchase based on their account balance.

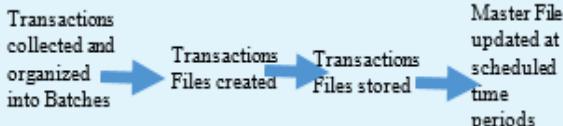
Batch and real-time transaction processing systems are the two types of transaction processing systems. Batch processing is a way of handling huge amounts of data in which a group of comparable transactions is grouped for a defined amount of time. It is commonly employed in manufacturing industries where large-scale manufacturing activities are carried out. A simple example of batch processing is when you do your own laundry. You will probably wait till you have a good amount to throw in the washer all at once.

Another example is when a consumer pays for a subscription service at the end of the month. Because the transactions happen at the same time, the TPS processes

them as a batch. It's fine if transactions take longer to process than they should as long as the system only reads batches once a month.

Real-time processing refers to data streams collected and processed without delay in real-time. Computer control is an example of this, in which a computer reacts instantly to events such as flight control, ATMs, and traffic control systems. Because data is processed as soon as it arrives, a continual flow of input data is required to offer a rapid output. Table 2.2 compares the batch and real-time processing.

Table 2.2: Batch vs Real-Time Processing

Batch processing	Real-Time processing
The collection and storage of data, for processing at a scheduled time when a sufficient amount of data has been accumulated.   <pre> graph LR     A[Transactions collected and organized into Batches] --&gt; B[Transactions Files created]     B --&gt; C[Transactions Files stored]     C --&gt; D[Master File updated at scheduled time periods]   </pre>	The immediate processing of data after the transaction occurs, with the database being updated at the time of the event.   <pre> graph LR     A[Transactions event occurring] --&gt; B[Online computer database updating]   </pre>
<b>Example:</b> (a) Cheque clearing, (b) Generation of bills, credit card transactions	<b>Examples:</b> (a) Reservation systems (b) Point of sales terminals
<b>Advantage</b> Many transactions are completed at once a time in a single process	<b>Advantage</b> Data is processed immediately
<b>Disadvantage</b> Data takes time to be processed	<b>Disadvantage</b> The act of processing data is repetitive.

### Management Information Systems

A Management Information System (MIS) is used to coordinate, control, analyse, and visualise information in an organisation. An example is a transportation coordinator using MIS reports to find the least and most expensive means that have been in use for the past year to ship products to market. MIS supports the management level of the organisation whereby managers can access various reports such as, performance reports, online. MIS usually depends on TPS for their data.

### Decision support systems

A decision support system (DSS) is an information system that aids in the decision-making process in an organisation. It assists people in making decisions regarding

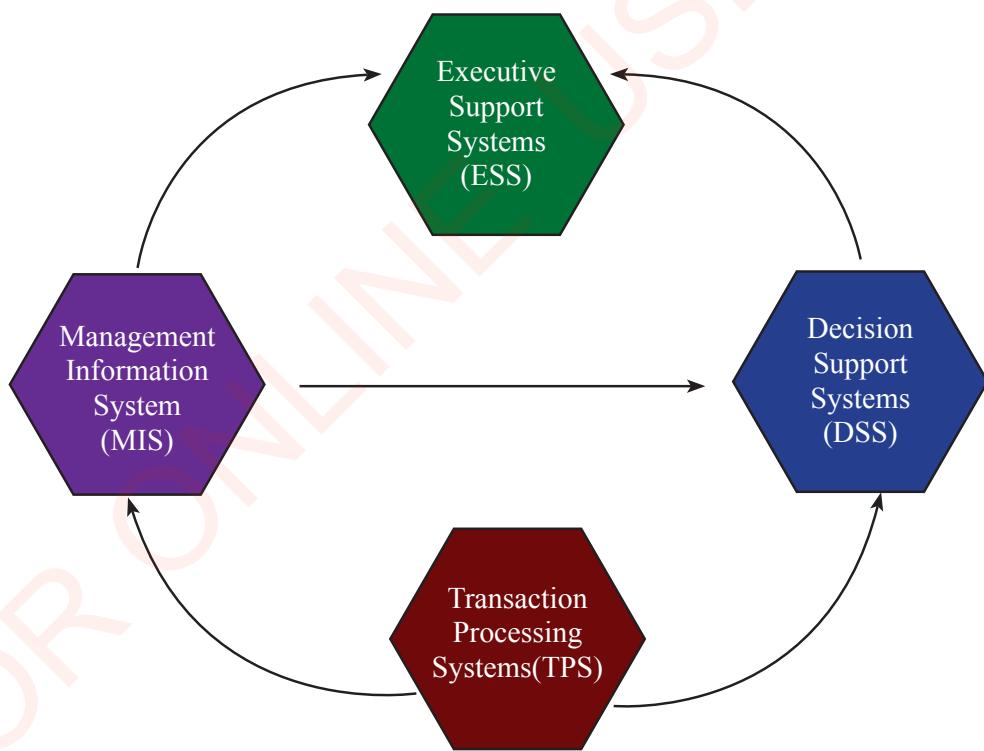
problems that are fast-changing and not clearly stated in advance, such as unstructured and semi-structured decision problems, at the management, operations, and planning levels of an organisation, often mid-to-upper management.

DSS focuses on combining model or analytic techniques with traditional data access and retrieval functions and emphasises flexibility and adaptability to accommodate environmental changes. It also focuses on features that make them easy to use by non-computer-proficient people in an interactive mode. Some examples are a sales manager projecting revenue figures based on product sales assumptions.

### Executive support systems

Executive Support Systems (ESS) or Executive Information Systems (EIS) are computer-based information systems that enable top managers or senior executives to quickly access internal and external data, critical to strategic decision-making, and other executive duties. It has drill-down features for easy information navigation, status access, trend analysis, exception reporting, and customised analysis, all of which are created with management's essential success elements in mind.

The four significant types of information systems are related to one another, as shown in Figure 2.8.



**Figure 2.8:** Major types of information systems and their interrelationships

TPS is typically a significant data source for other plans, whereas ESS is primarily a recipient of data from lower-level systems. Other types may exchange data with each other. Data may also be exchanged among systems serving different functional areas. For example, an order captured by a sales system may be transmitted to a manufacturing system as a transaction to produce or deliver the product specified in the order or to MIS for financial reporting.



### Activity 2.3: Analysing the types of information system

Eescribe TPS, DSS, ESS, and MIS and their interrelationship using an example of an information system from any organisation in your community or via a search engine.

### Exercise 2.2

- What is the importance of information systems in schools and for entrepreneurs today?
- What is the difference between batch and real-time processing in information systems? Provide an example for each.
- What roles do people play in information systems? How is it complemented by the procedures component of the information system?
- Consider the NECTA examination system accessible at [www.necta.go.tz](http://www.necta.go.tz), which you have probably used to access your form IV results. Provide an example of what can be done and a report obtained by each type of information system covered.

- (a) Provide an example of what can be done and identify the types of information system being used by necta for examination results.
- (b) What features do you think NECTA examination system possesses to enable managerial and decision making?

### Database systems

In today's world, database systems play a critical role in various aspects of our daily life, e.g., business and education. Every day, most of us and the organisations around us engage in various activities, including some form of database interaction. Such activities are depositing or withdrawing money from banks or mobile money agents, hotel reservations, buses, airline ticketing, and a digital library to access a bibliographic item or buy something online. Even buying groceries, today's supermarkets located across the country frequently update the database that stores the inventory of grocery goods. As a result, databases are practically vital in every field where computers are utilised, including business, government services, commerce, social media, engineering, healthcare, genetics, law, education, and library science.

A database is a collection of connected data that is organised and kept so that information may be retrieved as needed. It is a collection of related data having an underlying meaning that represents the reflection of the real world. We use the term "data" to refer to known facts that can be recorded and have implicit meaning. Consider the names, phone numbers, and addresses of the people you know. These days, most of this information is saved on mobile phones, which come with their database

software. This information can also be entered into an indexed address book or saved to a hard drive using a computer and software like Microsoft Access or MySQL.

A database is also a logically coherent collection of data with some inherent meaning that has been created, built, and populated with data for a particular purpose. It has a target audience and some predetermined applications that these users are interested in. In other words, a database has a data source, some level of interaction with real-world events, and an audience that is actively engaged in its contents. A database's end users can conduct transactions or an event can be triggered by an employee. All of this must be reflected in the appropriate databases.

A database must be a true reflection of the real world; it depicts to be accurate and dependable at all times. As a result, even in databases with increased size and complexity, such as the government electronic Payment Gateway (GePG), Facebook, and WhatsApp, which have millions to billions of users and are constantly changing, changes must be reflected in the database as soon as they occur. A database can range in size from an individual's address book to a company-wide database containing customer data used by employees to a product database used in conjunction with an e-commerce site to allow online shoppers to place real-time orders. Database can also be used in a search engine database. For example, Google contains data about billions of web pages accessed by people worldwide.

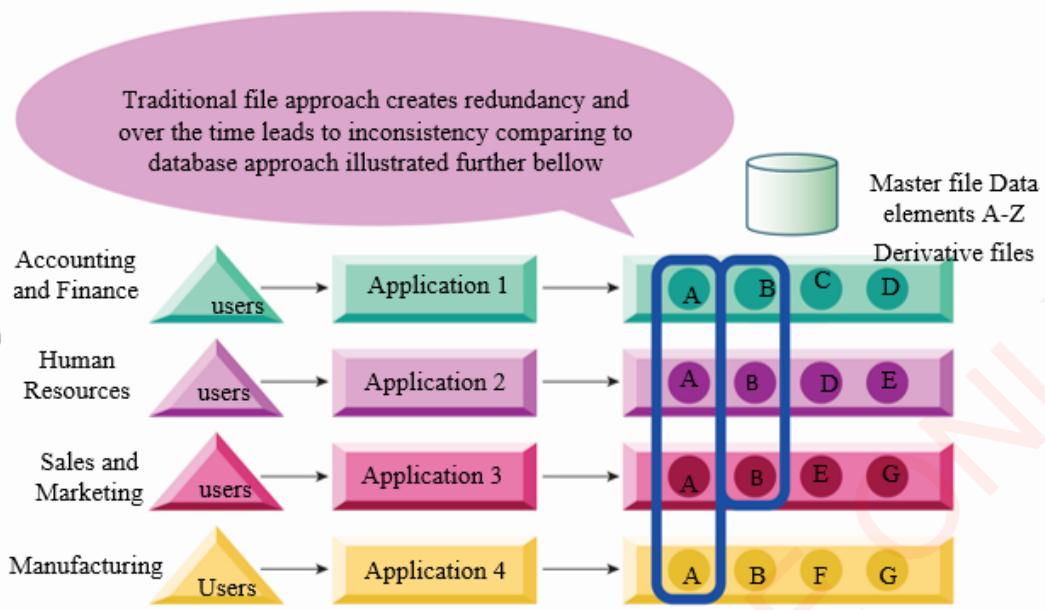
### Database system and file-based system

The file-based system is the software that allows users to access and organise a small group of data. It is defined in the

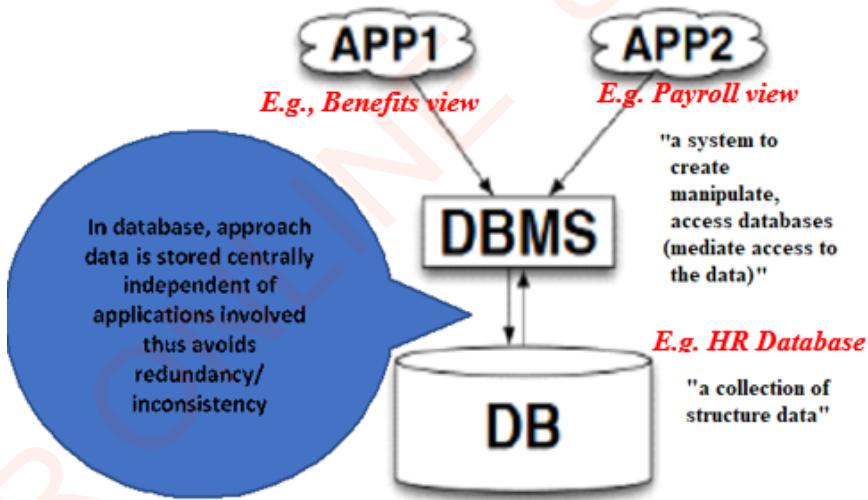
application manuals; so, programs are written to perform particular functions/applications. The database approach collects pragmatically related data, some of which can be published for an application. The main differences among them are as follows, and are also, illustrated in Figure 2.9:

- A file-based system is a method of arranging /organising files on a medium such as hard disks and flash disks. A Database system also known as Database Management System (DBMS) is a software application used to access, create, and maintain databases.
- There is no disaster recovery mechanism in the file-based system; DBMS provides you with a disaster recovery mechanism.
- The file-based system does not provide transfer of transactions; it is complex, while in the DBMS system, it is easy to implement the transfer operations using Structured Query Language.
- The file-based system does not offer concurrency, but DBMS does offer concurrency.
- Data redundancy and inconsistency are at the top of the system file. As shown in the first part of Figure 2.9(a), the same data, A, is stored in different application programs, thus causing redundancy which leads to data inconsistency as data get updated. The inconsistency in database management systems is eliminated or reduced when using different applications to access the same database, as shown in Figure 2.9(b). This is because all data are stored in one place (DB), independent of the applications involved and accessible by any of them. Also, any update made

through any application is stored centrally, and therefore, available for access by any application involved.



(a): *The traditional file approach*



(b): *The database approach*

**Figure 2.9:** (a) Traditional file system  
(b) The database approach

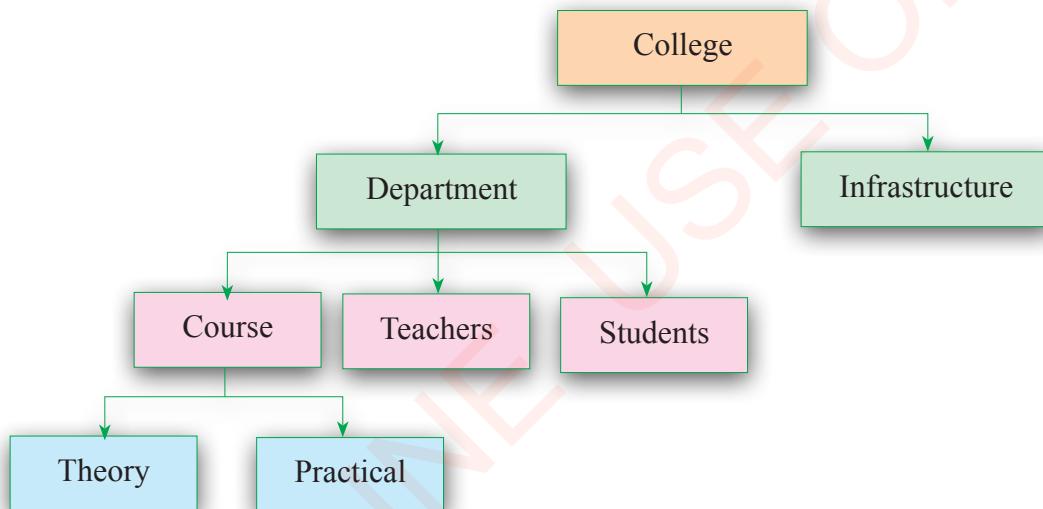
## Types of database models

A database model is a form of the data model that defines a database's logical structure. The model determines relationships and how data can be stored, organised, altered, and accessed. You can handle your business data using one of four database models: hierarchical, network, object-oriented, or relational database models.

### Hierarchical database model

A hierarchical database model is a data model that organises data into a tree-like

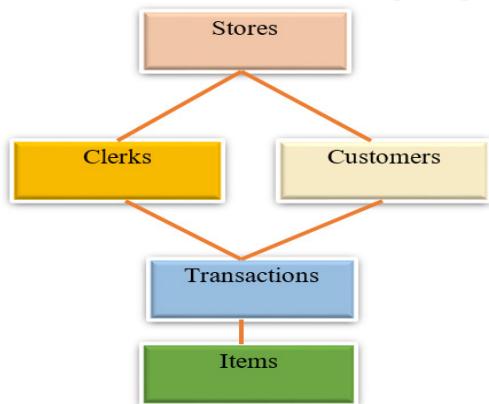
structure with a single root to which all other data is related. The information is kept in the form of records that are linked together. This model begins with the root data and grows in a tree-like fashion, adding child nodes to parent nodes, with each child node having only one parent node. This model may effectively express various real-world relationships, such as a book's index and recipes. This model also has a one-to-many link between two categories of data. For example, one department can have many courses, teachers, and students, as shown in Figure 2.10.



**Figure 2.10:** An example of a structure illustrating the use of the hierarchical database model

### Network database model

The network database model is designed to flexibly represent objects and their relationships. Data is organised more like a network in this model, and many parent nodes are allowed as shown in Figure 2.11. Network model extends the hierarchical structure since it provides many-to-many relationships to be managed in a tree-like structure with multiple parents. Because more associations are developed in this database model as the data become increasingly connected, accessing the data becomes quicker and faster, and redundancy is managed more efficiently than in the hierarchical approach. Because of this, the network database model was the most extensively used database model before the introduction of the relational model.



**Figure 2.11:** An example of a network database model

A relational database model represents data in which all items are organised into two-dimensional tables known as relations. It is currently the most extensively used model. In this paradigm, all data are kept in rows and columns in a table. Thus, compared to network and hierarchical models, the basic structure of a relational database model is simple, and it is readily scaled up because we may add as many rows and columns as needed. For example, in the student table, shown in Table 2.3 you can see columns (attributes or fields) and rows (tuples or records).

**Table 2. 3:** The attributes and tuples of a student table

Relation Name		Attribute (column)						
STUDENT		Name	Student_ID	Home_phone	Address	Office_phone	Age	Gpa
Juma Rashid	120-88-3221	0773663288	123 Kusini	0772009923		25	3.2	
Chausiku Mawazo	123-78-7689	0788712676	1454671 DSM	NULL		18	2.7	
Anna Majuto	888-08-7635	0625675412	543 Arusha	NULL		19	3.5	
Kulwa Doto	762-87-2596	0754431298	45 Kahama	0713988776		27	3.1	
Amina Karatasi	782-90-8754	0652870657	1276 Pemba	0772896514		23	4.1	

### Object-Oriented database model

In contrast to relational databases, which are table-oriented, an object-oriented database architecture represents both the data and the relationship in a single structure called an object. The object-oriented data model, better represents real-world problems. It allows us to store music, video, photographs, and other data types in databases, which was not possible or efficient in the relational paradigm. Furthermore, in this object-oriented database architecture, two or more objects with various properties and methods are linked together, as in today's object-oriented databases and programming languages. These links are used to connect one thing to another. Figure 2.12 shows an example in which each object has its properties and methods. For example, consider the employee object, "Salary" is one of the employee object properties, while "Get hired" is one of the methods.

Employee	Department
Attributes	Attributes
Name	Dept_id Name
Job_Tittle	Dept_name
Phone_no	Methods
Salary	Change Department
Dept_id	
Methods	
Get_Hired	
Change Number	

**Figure 2.13:** An example of object-oriented database model implementation



### Activity 2.3: Database systems

Access a file and a database system in your school and then describe the disadvantages observed in the former vs. advantages observed in the latter.

### Exercise 2.3

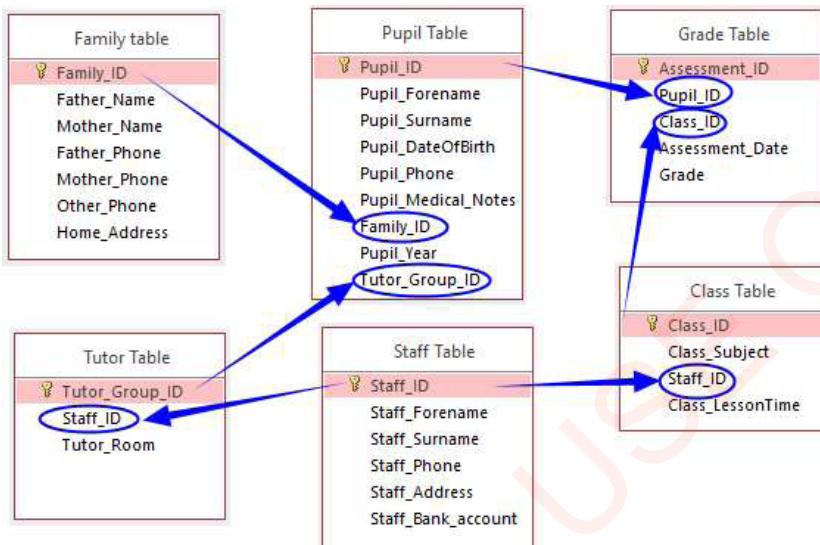
- What is the database system? How does it differ from the file system? Provide an example for each.
- Define the database model and describe its importance in the use of information systems today.
- Differentiate hierarchy and network models and then, provide real-life examples for each of them.
- Contrast relational and object-oriented models, and then give examples of their uses in real life for each of them.

## Relational database

### Meaning of relational database

A relational database is a collection of data elements linked together by pre-defined connections. These objects are organised as a collection of tables containing columns and rows, as mentioned in the preceding section. The relational database divides data into tables that can be connected or related based on shared data. You may create a whole new table from data in one or more existing tables with a single query. It also provides a better understanding of the relationships between all accessible data and the acquisition of fresh insights for better decision-making or identifying new opportunities. Consider the following scenario whereby your school keeps a record of the pupils and their related information, e.g., family members, results, subjects, and teachers, as shown in Figure 2.13. Pupil\_ID, Pupil Name, Family\_ID, and so on are examples of columns (or fields) in a pupil table; Assessment\_Date, Pupil\_ID, Grade, Class ID, and so on are examples of columns (or fields) in a grade table.

The tables (student and grade) can be linked together using the Pupil\_ID column. The linking between tables using common fields can also be applied to other tables in this small database. As a result, you can query the data to generate useful reports, such as consolidated pupil results. These queries are fed into report generators, which run them on-demand to create reports. Many organisations use databases to keep track of inventories, sales, financing, fees, treatments, and pharmaceuticals issued and develop financial and other business/organisational projections. This section will cover various aspects of relational databases, mainly entity-relationship models, functional dependency, and normalisation.



**Figure 2.13: Simple relational database example – Pupil Database**

### Relational database model

A relational database model is a collection of data items with pre-defined relationships between them, made possible by data models. Data models, which range from conceptual to logical, then to physical, serve as a powerful template and reference for your database. A data model is a set of specified data structures and business rules. It determines what data is required and how it should be organised, allowing for creating a database to be used to develop an application. They also use naming conventions and default values to assure data quality. It shows how different data pieces are connected to

generate a visual representation of data. It also responds to the data pieces' about who, what, where, and why questions. The data model in a retail transaction, for example, includes information on who made the purchase, what was purchased, and when. Additional information on the customer, product, store, salesperson, manufacturer, supply chain, and other factors could be included in the model. It is advisable to start with, high-level user needs (conceptual data model) and work your way through the middle level (logical data model) down to a granular database or file structure (physical data model). Three primary types of data models can be used to construct an

active database for various purposes. These models are conceptual, logical, and physical data models.

### **Conceptual data model**

The concepts, rules, and processes required to support the business are described using a conceptual data model. It determines and defines what data is needed for business operations, analytics, reporting applications, and the business rules and concepts that go with it. The conceptual model identifies the data required and tracks business events and their outcomes. On the other hand, a conceptual model concentrates on the necessary data pieces for business processes but ignores process flow or data type categorization.

Characteristics of a conceptual data model

- (a) The conceptual model is independent of hardware specifications like data storage capacity, and location, or software specifications like DBMS vendor and technology. The attention is to represent data in a way a user will see it in the “physical world.”
- (b) Gives organisation-wide coverage of the business concepts.
- (c) This type of data model is designed and developed for a business audience.

### **Logical data model**

A logical data model defines the mapping of data rules and structures. It goes into a greater depth about a process and how it ties to other business processes inside the company. It detects and defines data structures such as entities, tables, attributes, columns, and structure-to-structure relationships such as primary

and foreign keys. The goal of a logical data model is to define the data rather than to implement it. It is not reliant on any particular database or file structure. It can be used in various databases, including relational and NoSQL systems, and an XML file.

*Characteristics of a logical data model*

- (a) They are designed and developed independently from the DBMS.
- (b) It describes data needs for a single project but can integrate with other logical data models based on the project's scope.
- (c) Data attributes/elements will have data types with exact precisions and lengths. This characteristic may also be used in physical data model.
- (d) Normalization processes to the model are applied up to the Third Normal Form (3NF).

### **Physical Model**

A physical data model is more concerned with data design and database implementation. It specifies how data is organised in a system's specific database or file structure. It is the physical representation of process data pieces collected by the entity for storage in a database management system (DBMS). The model considers the technical as well as the performance criteria. The physical (model) comprises all relationships and entities, data types, keys, foreign keys, constraints, indexes, triggers, tables, spaces, partitions, and other DBMS features in the database utilised to construct an application. This model, developed from the conceptual and logical models, contains entities and relationships that help with enterprise data storage and retrieval.

## FOR ONLINE USE ONLY

*Characteristics of a physical data model* DUPLICATE

- (a) The physical data model describes data needed for a single project or application though it may be integrated with other physical data models based on project scope.
- (b) It was developed for a specific version of a DBMS, location, data storage, or technology in the project.
- (c) The physical data model contains relationships between tables that address cardinality and the relationships' nullability.
- (d) Columns should have exact data types, lengths assigned, and default values.
- (e) Primary and foreign keys, views, indexes, access profiles, authorizations, are defined.

The three data models, also shown in Figure 2.14, can be produced using Entity-Relationship modelling, a best practice for building a well-designed relational database structure through the three processes.

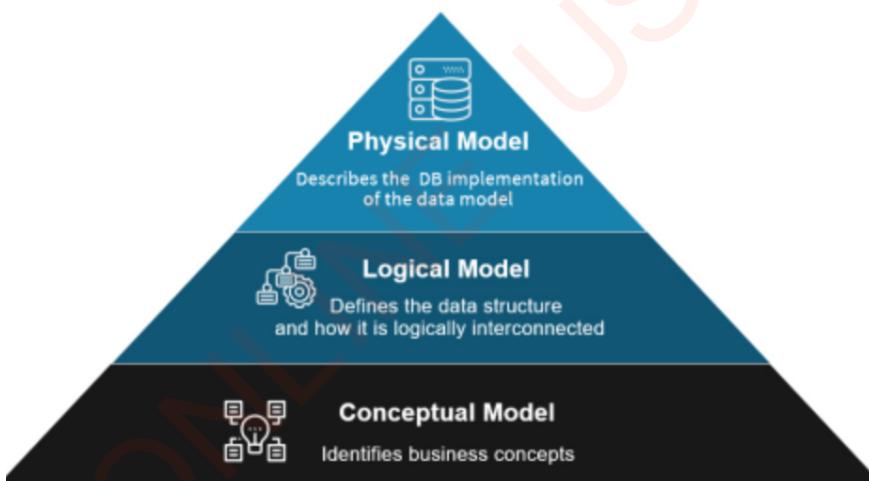


Figure 2.14: Conceptual, logical, and physical data models

Entities are objects that represent things in the real world and have a common set of attributes, (which are characteristics of properties of the entities), and relationships, (which describe how entities are related to one another, and share information in the database) are used to accomplish this database.

## Example on how to create the three data models from a real-life scenario

Scenario: A route taken by a particular bus company, whose buses travel according to a given schedule.

### Step 1: Creating a conceptual data model

In this step, you have to identify entities and their relationships. There are three entities, which are: Route, Bus Company, and Schedule. With regard to relationships, one route can have one or many bus companies offering transportation services to passengers. Also, a route can have one or multiple transportation schedules.

### Step 2: Creating a logical data model

Identification of characteristics, primary keys, and foreign keys are necessary for the establishment of a logical data model in addition to the names of entities and relationships that were formed between entities in step 1.

#### (i) Identification of attributes

- The entity Route may have the following attributes: *route number, route name, origin, destination*
- The entity Bus Company may have the following attributes: *company id, route number, owner name, post code*
- The entity Schedule may have the following attributes: *schedule number, route number, departure time, arrival time*

#### (ii) Primary keys

Under current scenario, the primary keys are route number, *company id*, and *schedule number* for entities Route, Bus Company, and Schedule, respectively.

#### (iii) Foreign keys

According to the relationships established in step 1, route number from the entity Route will become a foreign key in the remaining entities, which are *Bus Company* and *Schedule*. This implies that there shall be an addition of one attribute (*route number*) in the *Bus Company* and *Schedule* entities.

### Step 3: Creating a physical data model

The creation of the physical data model requires defining the names of the tables, names of the columns, and column data types in addition to the characteristics, primary key, and foreign keys described in the logical data model in step 2. Since physical data model is also an implementation of the database, its specification may be dependent on a particular database management system such as MS Access, MySQL, and PostgreSQL.

#### i) Names of tables

The names of tables are *route*, *bus\_company*, and *schedule*

#### ii) Names of columns

- The table *route* has the following columns, *route\_number, route\_name, origin, destination*
- The table *bus\_company* has the following columns, *company\_id, route\_number, owner\_name, post\_code*
- The table *schedule* has the following columns, *schedule\_number, route\_number, departure\_time, arrival\_time*

#### (iii) Data types

- Data type for *route\_number, company\_id, schedule\_number, and post\_code* will be integer

- Data type for *route\_name*, *origin*, *destination*, and *owner\_name* will be character of varying length (preferably 50 characters)
- Data type for *departure\_time*, and *arrival\_time* will be date time or time (depending on your preference)

relation, domain, primary key, candidate key, alternate key, foreign key, selection, and projection.

#### **Attribute or field**

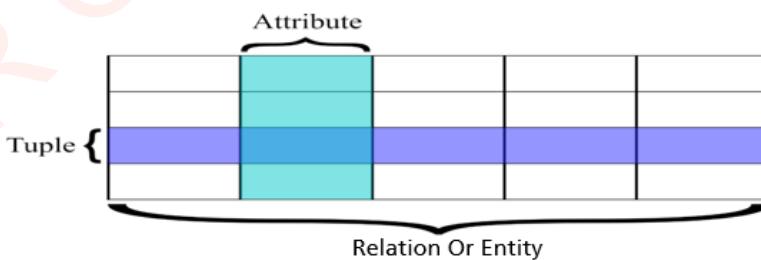
Attributes or fields describe characteristics or properties that define all items about a particular category. An attribute is also a named column in a database. e.g., first name, postal code, and date of birth in the employees' database.

#### **Tuple or record**

A tuple or record is a data set representing a single item. It is a collection of related data fields or attributes. It is also known as a row in a database. An example of a record is, for instance, when you have an employee's database with attributes *first\_name*, *post\_code*, and *date\_of\_birth* whose values may be Juma, 11101 Kivukon, and 2/3/1980, respectively

#### **Relation**

It is a set of tuples or records with the same attributes; a set of columns and rows as in Figure 2.15. Also, we refer to it as an entity that is a generalised class of people, places, concepts, or things for which data is collected, stored, and maintained. The entity is also named Table in a database. For example, a school database may include Students, Classes, and Courses.



**Figure 2.15:** Illustration of some relational database basic terminologies

## Domain

The domain is a set of allowable values for an attribute. It refers to a group of values we can assign to an attribute and can be considered as a constraint on the attribute's value. Each attribute has a domain. For example, for an attribute "Grade" in a student database, its domain can be a positive number between 0 and 100. Another example could be an attribute "Regions" in the Tanzania census database in which by March 2022, there were 31 regions (26 from Tanzania mainland and 5 from Zanzibar); thus, the domain alphabetically is "Arusha, Dar es Salaam, Dodoma ..., Kaskazini Pemba, Kusini Unguja ..., Tabora, Tanga".

## Primary key

The primary key is a tuple, field, or set of tuples or fields that uniquely identify a record or tuple in relation to a table. A table can have only one primary key, often denoted as PK. The primary key cannot be null due to the dependency of other fields in relation to a table. A good example is a National Identification Number in National Identification Authority (NIDA) database or a student registration number in a student database, as indicated in Figure 2.16 as "StudID."



**Figure 2.16:** Example of the primary, candidate and alternate keys using a student table

## Candidate key

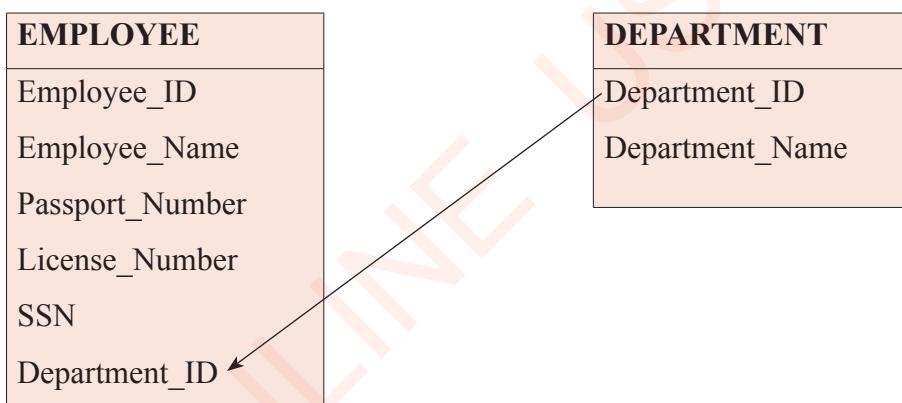
A candidate key, denoted as K, is a minimal set of attributes that can uniquely identify a tuple, or field row. Any set of attributes or columns has a unique combination of values in each row. The additional constraint is that removing any attribute or column can produce duplicate rows. The difference between the candidate key and the primary key is that, while the primary key uniquely identifies a record or row in the table and follows one key per table approach, the candidate key does the same. Still, it can have one or many keys per table. Sometimes, there are a group of columns in a table with unique values. All such columns have a candidate key, and thus the primary key is a subset of the candidate key. For example, in Figure 2.17, the StudID, Roll\_Number, and email columns can be the candidate keys, meaning records of these columns do not have any duplicate value, and they can be used to identify data in the table. And any one of these three can be taken as the primary key.

**Alternate key**

The candidate keys that are not selected as the primary key are called alternate keys. In this kind of key, columns have a candidate key except for the primary key, which has an alternate key. For example, in Figure 2.17, the columns - Roll\_No and Email are alternate keys, considering that Student ID is the primary key.

**Foreign Key**

The foreign key is an attribute, or set of attributes, within one relation or table that matches the primary key of another relation or table. It creates a relationship between the two tables. It effectively uses the values of attributes in the referenced relation to restrict the domain of one or more attributes in the referencing relation. The purpose is to maintain data integrity and allow navigation between two different instances of an entity. A good example is a foreign key in an Employee database where every employee works in a specific department in an organisation, and the employee and department are two different entities or tables. Therefore, we cannot store the department's information on the employee table. That is why we link these two tables through the primary key of one table. We add the primary key of the DEPARTMENT table, Department\_ID, as a new attribute in the EMPLOYEE table; thus, becoming a foreign key there; then both tables are related, as depicted in Figure 2.17.



**Figure 2.17:** Example of the foreign key using an employee database



### Activity 2.6. Identify keys in the tables

Consider the related tables of student registration and results in your school. Identify the Primary, Foreign, Candidate, and Alternate keys and illustrate them with a diagram.

## Composite Key

In the context of relational databases, a composite key is a combination of two or more columns in a table that can be used to identify each row in the table uniquely. Uniqueness is only guaranteed when the columns are combined; the columns do not guarantee uniqueness when taken individually.

The column(s) that agrees with uniqueness is called a candidate key. A composite key is a distinct type of candidate key formed by two or more columns. Sometimes the candidate key is just a single column, and sometimes it is created by linking multiple columns.

### Note:

- (a) The combination can also make a composite key of more than one candidate key.
- (b) A composite key cannot be empty/null.

### *Selection and projection*

Selection and projection operations are basic and most used operations specific to relational databases. The selection is defined as taking the horizontal subset of rows of a single table that satisfies a particular condition. It retrieves tuples from a relation, limiting the results to only those that meet a specific criterion. Selection, covered in detail in the next section, returns some or all rows and all the columns in a table. Selection answers the question: which rows shall the query return (where condition)? For example, 'Select \* From emp Where deptno = 10' as shown in Table 2.4.

**Table 2.4:** Selection operation in a relational database

EMPNO	FNAME	JOB	HIREDATE	SAL	DEPTNO
7782	Juma	Manager	09-june-2001	2,800,000	10
7839	Asha	Data Clerk	17-nov-2001	950,000	10
6289	Mariam	Analyst	23-jan-2002	2,250,000	10
5234	John	Data clerk	02-apr-2001	975,000	20
5235	Pima	Analyst	03-dec-2001	2,450,000	30
3255	Asha	Director	23-may-2007	4,000,000	40
2646	Zuhura	Manager	20-may-2010	3,000,000	20
5433	Peter	Ass. Director	30-jan-2009	3,800,000	20
2555	Jim	Analyst	06-sept-2011	2,100,000	30
2434	Shukran	Data clerk	02-oct-2010	950,000	30
9546	Hashim	Saleperson	23-may-2007	2,500,000	20
4356	Sura	Salesperson	06-sept-2011	1,500,000	30
4666	Busara	Manager	02-aug-2010	2,650,000	20
7677	Ansel	Ass. Director	23-may-2007	4,000,000	20

The projection is defined as the set obtained when the components of the tuples are restricted to the set – it discards (or excludes) the other attributes. It extracts only the specified relation attributes from a tuple or set of tuples. It is a vertical subset of table columns, and it may include one column, two columns or all table columns.

## FOR ONLINE USE ONLY

© 2017 TANZANIA INSTITUTE OF EDUCATION

Projection answers the question: which columns shall the query return? E.g. `Select empno, fname, job From emp'; returns columns as depicted in Table 2.5.

**Table 2.5:** Projection operation in the relational database

EMPNO	FNAME	JOB	HIREDATE	SAL	DEPTNO
7782	Juma	Manager	09-june-2001	2,800,000	10
7839	Asha	Data Clerk	17-nov-2001	950,000	10
6289	Mariam	Analyst	23-jan-2002	2,250,000	10
5234	John	Data clerk	02-apr-2001	975,000	20
5235	Pima	Analyst	03-dec-2001	2,450,000	30
3255	Asha	Director	23-may-2007	4,000,000	40
2646	Zuhura	Manager	20-may-2010	3,000,000	20
5433	Peter	Ass. Director	30-jan-2009	3,800,000	20
2555	Jim	Analyst	06-sept-2011	2,100,000	30
2434	Shukran	Data clerk	02-oct-2010	950,000	30
9546	Hashim	Saleperson	23-may-2007	2,500,000	20
4356	Sura	Salesperson	06-sept-2011	1,500,000	30
4666	Busara	Manager	02-aug-2010	2,650,000	20
7677	Ansel	Ass. Director	23-may-2007	4,000,000	20

Projection differs from selection as SELECT is used to select all rows of a specific relation that meet a particular condition, whereas project is used to select particular columns. However, one query may contain both: selection and projection: For example, `select empno, fname, job From emp Where deptno=10'; returns enclosed columns and rows as in Table 2.6.

**Table 2.6:** Selection and Projection operations in a relational database

EMPNO	FNAME	JOB	HIREDATE	SAL	DEPTNO
7782	Juma	Manager	09-june-2001	2,800,000	10
7839	Asha	Data Clerk	17-nov-2001	950,000	10
6289	Mariam	Analyst	23-jan-2002	2,250,000	10
5234	John	Data clerk	02-apr-2001	975,000	20
5235	Pima	Analyst	03-dec-2001	2,450,000	30
3255	Asha	Director	23-may-2007	4,000,000	40
2646	Zuhura	Manager	20-may-2010	3,000,000	20
5433	Peter	Ass. Director	30-jan-2009	3,800,000	20
2555	Jim	Analyst	06-sept-2011	2,100,000	30
2434	Shukran	Data clerk	02-oct-2010	950,000	30
9546	Hashim	Saleperson	23-may-2007	2,500,000	20
4356	Sura	Salesperson	06-sept-2011	1,500,000	30
4666	Busara	Manager	02-aug-2010	2,650,000	20



### Activity 2.7: Selection and projection

Using selection and projection based on Table 2.6, work out the results of “Select Fname, job, hiredate, sal From emp Where deptno=30”. Illustrate the results in the same format as in the referred figure.

## *Advantages of relational database model*

A relational database system has multiple advantages, as described in the next paragraphs.

### *Simple model*

A relational database system is the simplest model, as it does not require any complex structuring or querying processes. It does not involve tedious architectural processes and SQL queries like hierarchical and network database models.

### *Data accuracy*

In the relational database system, multiple tables can be related to one another using a primary key and foreign key concepts. This makes the data non-repetitive hence, no data duplication, and more data accuracy is experienced.

### *Easy access to data*

There is no pattern or pathway for accessing the data in the relational database system as it is in the network or hierarchical models. Anyone who accesses the data can query any table or a combination of them in the relational database, allowing the user to get and display the relevant data effortlessly.

### *Data integrity*

Data integrity is a crucial characteristic of the relational database system. Robust data entries and legitimacy validations ensure that all the data in the database is confined within suitable arrangements and the necessity for creating the relationships are present. This relational reliability amongst the tables in the database helps to prevent

the records from being imperfect, isolated or unrelated; thus, aiding the relational database's characteristics like ease of use, precision, and data stability.

### *Flexibility*

A relational database system possesses qualities for levelling up and expanding to bigger lengths. It is endowed with a bendable structure to accommodate the constantly shifting requirements, as there is supposedly no boundary on the number of rows, columns or tables a relational database can hold. This facilitates the increasing incoming amount of data and the update and deletes wherever required. For example, a database administrator can insert, update, or delete tables, columns, or individual data in the given database system promptly and efficiently to meet business needs.

### *Normalization*

The methodical style is maintained to make sure that a relational database structure is liberated of any variances that can make a difference in the integrity and accuracy of the tables in the database. A normalization process provides a set of regulations, characteristics, and purposes for the database structure and evaluation of a relational database model. It aims at illustrating multiple levels of breaking down the data. Any level of normalization is expected to be accomplished on the same level, that is, before moving ahead to the next levels. A relational database model is usually confirmed to be normalised only when it satisfies the third normalisation form's necessary conditions; thus, reassuring the extra strong and reliable database.

**High security**

As the data is divided amongst the tables of the relational database system, it is possible to make a few tables to be tagged as confidential and others not. Unlike other databases, this segregation is easily implemented with a relational database management system. For example, when a data analyst tries to log in with a username and password, the database can set boundaries for their level of access by providing admission only to the tables that they are allowed to work on.

**Feasible for future modifications**

As the relational database system holds records in separate tables based on their categories, it is straightforward to insert, delete or update records subject to the latest requirements. This feature of the relational database model tolerates the newest requirements presented by the business and can be inserted or modified depending on the conditions.

**Examples of relational database systems**

Relational database systems are used in many organisations today. There are several common and popular relational database systems based on: end-users, open source, and proprietary, as follows:

**Popular DBMSs for end-users**

The popular and most used relational database in this end user's utilisation category is Microsoft Access and File MakerPro.

**Popular open-source DBMSs for organizations**

The popular and most used relational

database in this category is MySQL and PostgreSQL.

**Popular proprietary DBMSs for organizations**

The popular and most used relational databases in this category are Microsoft SQL Server, Oracle, IBM-DB2 and recently AWS - Amazon Relational Database Service (Amazon RDS), that makes it easy to set up, operate, and scale a relational database in the cloud.

**Exercise 2.5**

1. What is a relational database model?
2. Differentiate between logical and physical data models.
3. Identify two related entities in your school students' organisation and determine three possible attributes for each. Create both tables with three tuples for each and show how they relate.
4. In the tables you have created in question (3), identify the primary key, candidate key, alternate key and foreign key. Also, determine the domain for each attribute that was identified.
5. Differentiate between selection and projection in an RDBMS while giving examples.

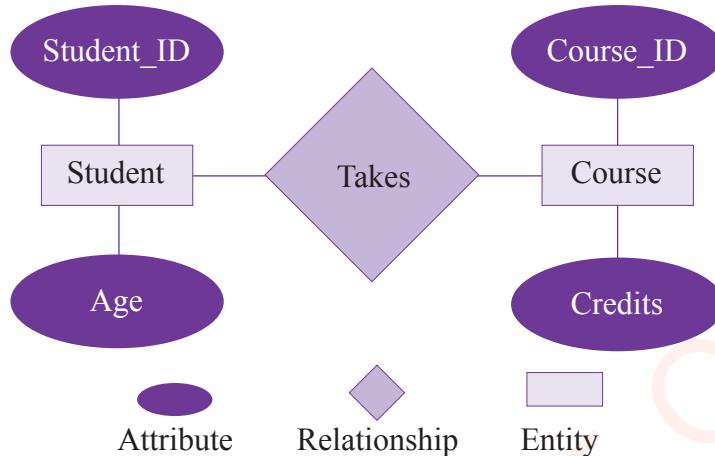
**Entity-relationship model**

Entity-Relationship (ER) modelling is a best practice for producing well-designed databases. It depicts the structure of a relational database allowing you to understand the data and how it shares information.

### *Elements of the entity-relationship model*

An ER data model's main elements involve entities, attributes, and relationships, as depicted in Figure 2.18,

with the example from a student database involving how each element is diagrammatically represented.



**Figure 2.18:** A sample of entities, attributes, and relationships from a student database in the ER data model

### **Entity**

An entity is an object representing things from the real world. An entity contains a set of attributes, each with values.

### **Attribute**

The attribute is the characteristics or properties of an entity based on the data type. Each column represents an attribute that stores the value of an entity characteristic.

### **Relationships**

Relationships define how entities are related to each other; and how they share information in the database. This is done through Entity-Relationship Diagrams (ERD). ERD uses basic graphical symbols to show the organisation of entities and their relationships. An important aspect of relationships is

cardinality which defines how the entities' data or records are related. The common cardinalities include one-to-one, one-to-many, many-to-many, and many-to-one relationships. Details for each relationship are outlined as follows:

*One-to-one relationship* is when one entity from entity set A can be associated with at most one entity of entity set B and vice versa. For example, a school might decide to assign one class teacher to precisely one class; thus, one teacher manages only one class.

*One-to-many relationship* is when one entity from entity set A can be associated with multiple entities of entity set B. Still, an entity from entity set B can be associated with at least one entity in set A. For example, the relationship between a teacher and the subjects they teach, where one teacher can teach

multiple subjects, but only one teacher summarised as follows:

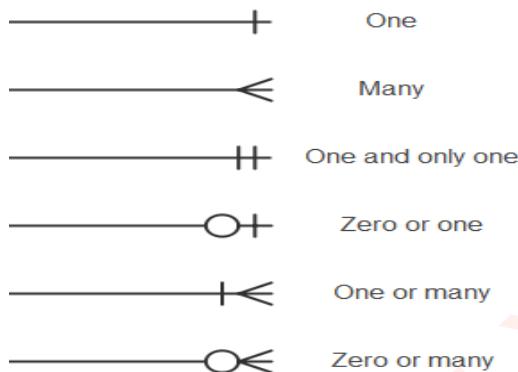
teaches a given subject.

*Many-to-one relationship* is when more than one entity from entity set A can be associated with at most one entity of entity set B. However, an entity from entity set B may or may not be associated with more than one entity from entity set A. For example, many students belong to the same class.

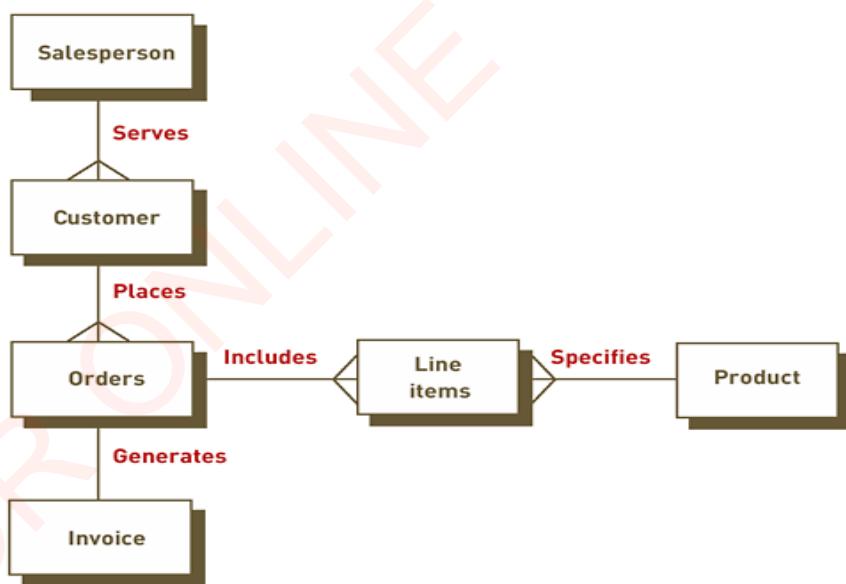
*Many-to-many relationship* is when one entity from set A can be associated with more than one entity from set B and vice versa. For example, students are related to multiple teachers, and teachers can be associated with multiple students.

### Cardinality notation symbols

In cardinality notation symbols, the lines connecting entities appear with extra marks at the end. These marks define the cardinality degree. The notations are



Generally, these cardinalities might apply in an ERD at the entity level. An example is the customer order database in Figure 2.19, in which a salesperson serves many customers that each can also place multiple orders. Each order can have only one invoice, but each of these orders may also include many line items, and each specifies products that it contains.

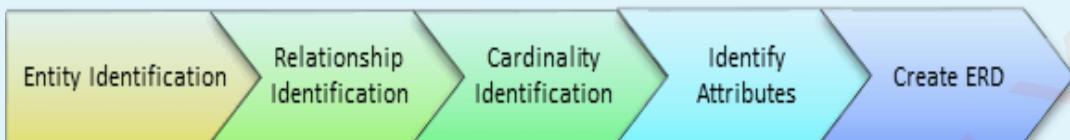


**Figure 2.19:** An example of Entity-Relationship Diagram (ERD) - customer order database



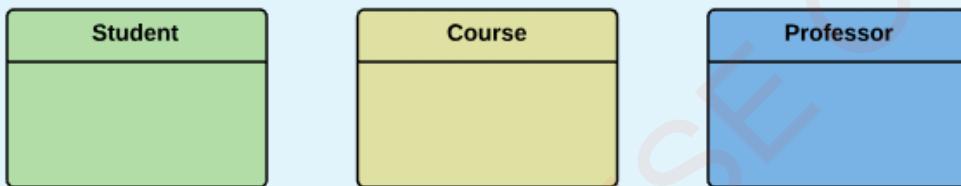
### Activity 2.8: Construction of an E-R diagram to represent a real-life database

In a University, a student enrols in courses. A student must be assigned to at least one or more courses. To maintain instruction quality, a Professor can deliver only one course. Create an Entity Relationship Diagram (ERD) following the steps indicated in the diagram below and the steps that follow:



#### Step 1) Entity identification

We have three entities: Student, Course and Professor.



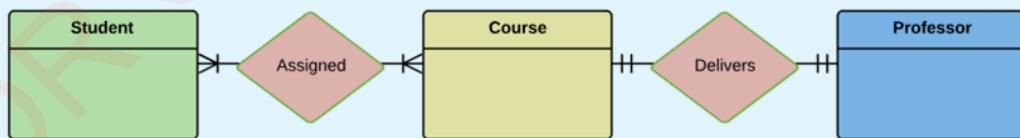
#### Step 2) Relationship identification

We have two relationships: The student is assigned a course, and the professor delivers a course.



#### Step 3) Cardinality identification

From the problem statement, we know that, a student can be assigned multiple courses, and at a time Professor can deliver only one course.



#### Step 4) Identify attributes

## FOR ONLINE USE ONLY

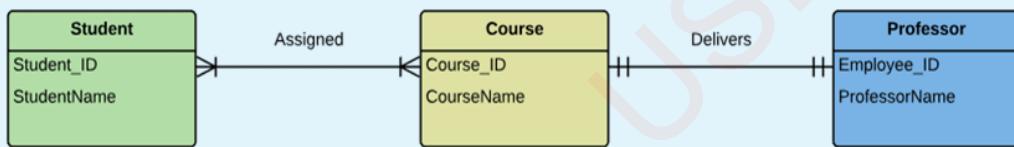
To identify attributes, you need to study the files, forms, reports, and data currently maintained by the organization. You can also conduct interviews with various stakeholders to identify entities. Initially, identify the attributes without mapping them to a particular entity. Once you have a list of attributes; you need to map them to the identified entities. Ensure an attribute is to be paired with exactly one entity. If you think an attribute should belong to more than one entity, use a modifier to make it unique. Once the mapping is done, identify the primary keys. If a unique key is not readily available, create one.

Entity	Primary Key	Attribute
Student	Student_ID	StudentName
Professor	Employee_ID	ProfessorName
Course	Course_ID	CourseName

For the Course entity, more attributes could for example, be Duration, Credits, Assignments, etc. For ease, we have considered just one attribute for each entity.

### Step 5) Create the ERD Diagram

A more modern representation of ERD example



### Functional dependency

#### *Meaning of functional dependency*

Functional dependency is a constraint that determines the relation of one attribute to another in a relational database management system. It is a relationship between attributes in which we can obtain the value of another attribute from a given attribute. For example, if we know the value of a student's roll number, we can obtain the student's address and marks. It is said that student address and marks are functionally dependent on student roll number. Similarly, if we know the value of the employee number, we can also obtain the employee name and salary;

thus, the employee name and salary are functionally dependent on the employee number. Table 2.7 shows an example of functional dependent.

**Table 2.7:** Functional dependent

Employee number	Employee Name	Salary
EMP01	Diana	900,000
EMP02	Juma	950,000
EMP03	Andrew	850,000

#### *Importance of functional dependencies*

- (a) Functional dependency helps to maintain the quality of data in the database.

- (b) It helps to define databases' constraints, ensure data integrity, and avoid data anomalies.
- (c) It plays a vital role in finding the difference between good and bad database design, thus promoting robust and impactful databases.

### Types of functional dependencies

There are four main types of functional dependency which are:

- (a) Multivalued dependency
- (b) Trivial functional dependency
- (c) Non-trivial functional dependency
- (d) Transitive functional dependency

#### *Multivalued dependency*

This type of functional dependency occurs when there are multiple independent multivalued attributes in a single table. It consists of at least two dependent attributes on its attributes; thus, it always requires at least three attributes. For example, consider a motorbike manufacturing company that produces two colours (black and red) in each model every year. Table 2.8 shows multivalued dependency.

Table 2.8: *Multivalued dependency*

Motorbike_model	Manuf_year	Colour
M1001	2017	Black
M1001	2017	Red
M2012	2018	Black
M2012	2018	Red
M2222	2019	Black
M2222	2019	Red

Here columns manuf\_year and colour are independent of each other and dependent on motorbike\_model. In this case, the two columns are said to be multivalued dependent on motorbike\_model. These dependencies are represented like this: motorbike\_model  $\rightarrow\!\!>$  manuf\_year; and motorbike\_model  $\rightarrow\!\!>$  colour

#### *Trivial dependency*

This type of functional dependency occurs when a set of attributes called a trivial is included in that attribute. So,  $X \rightarrow Y$  is a trivial functional dependency if Y is a subset of X. For example, consider this Employee table with two columns, Emp\_id and Emp\_name.

Emp_id	Emp_name
AS555	Harry
AS811	Mwajuma
AS999	Kevin

In this case, {Emp\_id, Emp\_name}  $\rightarrow$  Emp\_id is a trivial functional dependency as Emp\_id is a subset of {Emp\_id, Emp\_name}.

#### *Non-trivial dependency*

This functional dependency occurs when A  $\rightarrow\!\!>$  B holds true where B is not a subset of A. In other words, if a functional dependency  $X \rightarrow Y$  holds true, then Y is not a subset of X; thus, this dependency is called a non-trivial functional dependency. For example, an employee table with three attributes: emp\_id, emp\_name and emp\_address. From this example, the following functional dependencies are non-trivial: emp\_id  $\rightarrow$  emp\_name (This means that if we know emp\_id, we know emp\_name. but emp\_name is not a subset of emp\_id) emp\_id

-> emp\_address (This imply that, if we know emp\_id, we know emp\_address which is not a subset of emp\_id).

### A transitive dependency

It is a type of functional dependency that happens when it is indirectly formed by two functional dependencies, and can only occur in relation to three or more attributes. Consider the following table containing the details of the company's Chief Executive Officers (CEOs).

**Table 2.9:** Company chief executive officers details

Company	CEO	BirthDate
Microsoft	Satya Nadella	1.2.1967
Google	Sundar Pichai	12. 6.1972
Facebook	Mark Zuckerberg	7.9.1984

It follows that {Company}  $\rightarrow$  {CEO} (if we know the company, we know its CEO's name). {CEO}  $\rightarrow$  {Birthdate}; if we know the CEO, we know the Birthdate. Therefore, if the rule of transitive dependency: {Company}  $\rightarrow$  {Birthdate} should hold, it will make sense because if we know the company's name, we can know the CEO birthdate.



### Activity 2.9: Functional dependencies in databases

The multivalued dependency is the type of functional dependency that occurs when multiple independent multivalued attributes are in a single table. Study your surroundings and generate a single table that produces a single table that is a multivalued dependency.

## Normalisation

### Meaning of normalisation

Normalisation was developed to break data into related groups and define the relationships between them. It is a specific relational database analysis and design technique used to model groups of related data within an organisation. Normalisation organises data in a database to avoid redundancy, insertion anomaly, update anomaly, and deletion anomaly. These concepts will be discussed in detail later in this section. It analyses the relation schemas based on their different functional dependencies and primary key. The purpose is to ensure that data stored within the database adhere to best practices by following a set of rules to eliminate redundancies and optimise the process of information retrieval. Normalisation leaves us with a structure that groups alike data into relational models. It is accomplished by applying some formal rules either by synthesis (creating a new database design) or decomposition (improving an existing database design).

### Importance of normalisation

Database normalisation is structuring a database, usually a relational database, to reduce data redundancy, anomalies, and inconsistency; and minimising redesign when extending the database structure. It also entails organising a database's columns (attributes) and tables (relations) to ensure that database integrity constraints properly enforce their dependencies. An example indicating one of the reasons for normalization, the data anomalies, is shown in Table 2.10, where Employee 426 is shown as having different addresses on different records.

Table 2. 10: Data update anomaly – employee 426 shows different addresses on its records

	Employees' Skills	
Employee ID	Employee Address	Skill
426	Box 137 MWANZA	Public speaking
426	Box 132 MWANZA	Typing
645	Box 8991 Dar es salaam	Carpentry
645	Box 8991 Dar es salaam	Public speaking

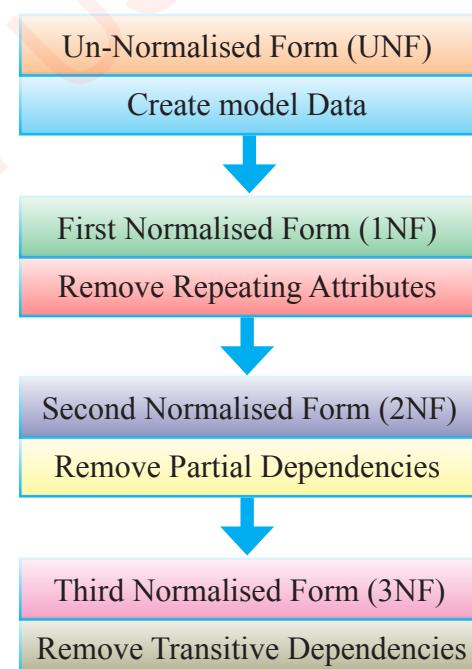
In this example, the same information can be expressed on multiple rows; therefore, updates to the relationship may result in logical inconsistencies. For example, each record in an “Employees’ Skills” relation might contain an Employee ID, Employee Address, and Skill; thus, a change of address for a particular employee may need to be applied to multiple records (one for each skill). If the update is only partially successful, (the employee’s address is updated on some records but not others), then the relation is left in an inconsistent state. Specifically, the relation provides conflicting answers to this particular employee’s address, affecting the database’s use and effectiveness.

### Database normalisation forms

Normalisation is represented by a logical set of steps that follow simple rules applied to each stage of the modelling process. The stages are separated into normal forms at the highest level, identified by a particular named process. There are three common normal forms that, when applied, a relation is often described as “normalised”. These forms are First Normal Form (1NF), Second Normal Form, (2NF) and Third Normal Form (3NF). Generally, normal forms are six, but the rest three are considered complex data structures. Once the first three common normal forms are applied,

relations are considered free of insertion, updating, and deletion anomalies.

In addition, we have something called Un-Normalised Form (UNF); though not generally considered as part of the normalisation rules, it represents the very first stages of the normalisation process. The objective of the Un-Normalised Form (UNF) and each of the three normal forms (1NF; 2NF; 3NF) are shown in Figure 2.20, followed by details for each one.



**Figure 2.20:** Objective of Un-Normalised Form (UNF) and the three normal forms (1NF; 2NF; 3NF)

***Un-Normalised Form (UNF)***

Un-Normalised Form (UNF) intends to achieve a database data model (organisation of data in a database) which does not meet any of the conditions of database normalisation defined by the relational model. In the relational model, unnormalised relations can be considered the starting point for normalisation. This un-normalised form is a preparatory stage of the normalisation process, allowing us to create a structured frame representative of organisational data such as a form or document (e.g., invoice, report and purchase order).

Based on the form or document you are working from, draw up a table structure (unnormalised), creating a column heading for each data item.

An example of the unnormalised form: is when a manufacturing company stores the employee details in a table named employee with four attributes, as shown in Table 2.11.

**Table 2.11:** An example of an unnormalised form table

emp_id	emp_name	emp_address	emp_dept
101	Samson Madirisha	DSM	D001
101	Samson Madirisha	DSM	D002
123	Madina Wavu	Kagera	D890
166	Hashim Milango	Mbeya	D900
166	Hashim Milango	Mbeya	D004

In this un-normalised table (see Table 2.11), there is an **update anomaly**. In the table, we have two rows for employee Samson, who belongs to two different departments of the company. If we want to update the address of Samson, then we have to update the same in two rows; otherwise, the data will become inconsistent. If, somehow, the correct address were updated in one department but not in the other, then as per the database, Samson would have two different addresses, which is incorrect and would lead to inconsistent data. Besides, the table in this form may lead to an **insert anomaly**. Suppose a new employee joins the company who is under training and currently not assigned to any department, then we would not be able to insert the data into the table if the emp\_dept field does not allow nulls. Finally, in this normalized table, there is a **delete anomaly**: suppose, at a point in time, the company closes the department D890, and then deleting the rows that have emp\_dept as D890 would also, delete the information of employee Madina since she is assigned only to this department. Therefore, we need to normalize the data in the three normal forms as introduced earlier to overcome these anomalies.

Another example of an Unnormarised Table is shown in table 2.12.

**Table 2.12:** Non-normalized table.

Full name	Address	Movie_industry	Sports
Ismail Makame	Kijichi	Hollywood and Bollywood	Football and Basketball
Rehema Kiongwe	Mnazi Moja	Swahiliwood and Nollywood	Netball
Muki Mohamed	Mbagala	Nollywood and Riverwood	Cricket

### Advantages of normalization

- (a) Normalization helps to reduce data redundancy
- (b) More incredible overall database organization
- (c) Data consistency within the database
- (d) Much more flexible database design
- (e) Enforces the concept of relational integrity

### Disadvantages of normalization

- (a) You cannot start building the database before knowing what the user needs.
- (b) The performance degrades when normalizing the relations to higher normal forms, i.e., 4NF, and 5NF.
- (c) It is very time-consuming and challenging to normalize relations of a higher degree.
- (d) Careless decomposition may lead to a bad database design and severe problems.

### First Normal Form (1NF)

As indicated in Figure 2.20, the objective of this 1NF is to remove repeating attributes. This is done because the relation is in 1NF if it contains an atomic value, and an attribute of a table cannot hold multiple values. Instead must have only single-valued attributes. Also, 1NF disallows the multi-valued attribute, composite attribute, and their combinations.

For a table to be in the First Normal Form, it should follow the following four rules:

1. It should only have single (atomic) valued attributes/columns.
2. Values stored in a column should be of the same domain.
3. All the columns in a table should have unique names.
4. The order in which data are stored does not matter.

For example: Suppose an agriculture-based company wants to store its employees' names and contact details. It creates a table as presented in Table 2.13.

**Table 2.13:** An example of a table not conforming to 1NF

emp_id	emp_name	emp_address	emp_mobile
101	Hamis Waya	Iringa	0658000770
102	Erick Misitu	Mbeya	0713876776 0767000771
103	Mwantumu Chake	Coastal	0688000775
104	Namdori Mchicha	Arusha	028675432 0736000671

As you can see from Table 2.13, employees (Erick and Namdori) have two mobile numbers, so the company stored them in the same field as you can see in emp\_mobile. Therefore, this table is not in 1NF as a rule says, “each attribute of a table must have atomic (single) values”; the emp\_mobile values for employees Erick and Namdori violate this rule.

To make the table comply with 1NF, we should have those anomalies fixed in the data, as indicated in Table 2.14.

**Table 2.14:** An example of a table conforming to the First Norm Form (1NF).

emp_id	emp_name	emp_address	emp_mobile
101	Hamis Waya	Iringa	0658000770
102	Erick Misitu	Mbeya	0713876776
102	Erick Misitu	Mbeya	0767000771
103	Mwantumu Chake	Coastal	0688000775
104	Namdori Mchicha	Arusha	028675432
104	Namdori Mchicha	Arusha	0736000671

For example, if we want to put Table 2.15 in the 1N, it will look as in Table 2.15.

**Table 2.15:** The first normalized form

Full name	Address	Movie_industry	Sports
Ismail Makame	Kijichi	Hollywood	Football
Ismail Makame	Kijichi	Bollywood	Basketball
Ismail Makame	Kijichi	Hollywood	Basketball
Ismail Makame	Kijichi	Bollywood	Football
Rehema Kiongwe	Mnazi Moja	Swahiliwood	Netball
Rehema Kiongwe	Mnazi Moja	Nollywood	Netball
Muki Mohamed	Mbagala	Riverwood	Cricket
Muki Mohamed	Mbagala	Nollywood	Cricket

**Second Normal Form (2NF) NOT DUF**

A table is in 2NF, if the relation or table is in 1NF (First Normal Form) and all non-key attributes are fully functional dependent on the primary key (non-prime attribute). For example, suppose a school wants to store the data of teachers and the subjects they teach and, in a school, a teacher can teach more than one subject. Since a teacher can teach more than one subject, the table can have multiple rows for the same teacher, as indicated in Table 2.16.

**Table 2.16:** An example of a table not conforming to 2NF

Teacher_id	Subject	Teacher_age
25	Computer	40
25	Biology	40
47	English	35
83	Math	36
83	Chemistry	36

Based on Table 2.16, we have; candidate keys which are {Teacher\_id, Subject} and the Non-prime attribute is: teacher\_age. Therefore, the non-prime attribute Teacher\_age is dependent on Teacher\_id, which is a proper subset of a candidate key; thus, violating the rule for 2NF. To convert the given table into 2NF, we decompose it into two tables: Teacher details in Table 2.17(a) and Teacher-subject in Table 2.17(b), which now comply with Second Normal Form (2NF). Resulting tables conforming to Second Norm Form (2NF).

**Table 2.17 (a):** Teacher details

Teacher_id	Teacher_age
25	40
47	35
83	36

**Table 2.17: (b)** Teacher\_subject table

Teacher_id	Subject
25	Computer
25	Biology
47	English
83	Math
83	Chemistry

The characteristics of 2NF

- (a) In the 2NF, relational must be in 1NF.
- (b) In the second normal form, all non-key attributes are fully functional dependent on the primary key.
- (c) A single column Primary Key that is not functionally dependant on any subset of candidate key relation.

**Third Normal Form (3NF)**

A table design is said to be in 3NF if it is in 2NF and transitive functional dependency of the non-prime attribute (*an attribute that is not part of any candidate key*) on any super key (*single key or a group of multiple keys that can uniquely identify tuples in a table*) should be removed.

In other words, a table can be explained as in 3NF if it is in 2NF, and for each functional dependency  $X \rightarrow Y$ , at least X is a super key of the table or Y is a prime attribute of the table. For example, suppose a company wants to store the complete address of each employee; they create a table named employee\_details, as indicated in Table 2.18.

**Table 2.18:** An example of a table not conforming to 3NF.

emp_id	emp_name	emp_pc	emp_reg	emp_city	emp_district
1001	John Miti	282005	AR	Arusha	Arumeru
1002	Muhidini Kata	222008	DS	DSM	Kinondoni
1006	Lora Misitu	282007	DS	DSM	Temeke
1101	Lilly Nyavu	292008	TG	Tanga	Muheza
1201	Steve Milano	222999	MZ	Mwanza	Ilemela

In Table 2.18 you can see that Super keys are: {emp\_id}, {emp\_id, emp\_name}, {emp\_id, emp\_name, emp\_pc}... and so on; Candidate Keys: {emp\_id}, Non-prime attributes: all attributes except emp\_id are non-prime as they are not part of any candidate keys.

Here, emp\_reg, emp\_city and emp\_district depend on emp\_pc. and, emp\_pc is dependent on emp\_id that makes non-prime attributes (emp\_reg, emp\_city & emp\_district) transitively dependent on super key (emp\_id). This violates the rule of 3NF. To make this table comply with 3NF, we have to break the table into two tables to remove the transitive dependency, as shown in Table 2.19 and Table 2.20, which conform to 3NF.

**Table 2.19:** Employees' details

emp_id	emp_name	emp_pc
1001	John Miti	282005
1002	Muhidini Kata	222008
1006	Lora Misitu	282007
1101	Lilly Nyavu	292008
1201	Steve Milano	222999

**Table 2.20:** Employee address

emp_pc	emp_reg	emp_city	emp_district
282005	AR	Arusha	Arumeru
222008	DS	DSM	Kinondoni
282007	DS	DSM	Temeke
292008	TG	Tanga	Muheza
222999	MZ	Mwanza	Ilemela



### Activity 2.9: Normalising databases up to the 3rd Normal form - Examine Table 2.17:

Study the following table and then answer the questions that follows:

**Table 2.21:** Employee details

staffNo	branch No	branchAddress	name	position	hr/week
S3244	B0021	Temeke, Plt 12G DSM	Asha Haji	Assistant	16
S3244	B0045	Mikocheni, Plt 32A, DSM	Asha Haji	Assistant	9
4545	B0021	Temeke, Plt 12GDSM	Mayalla Macho	Assistant	14
S4545	B0045	Mikocheni, Plt 32A, DSM	Mayalla Macho	Assistant	10

- Why is this table not in 3NF? Give two reasons.
- Describe and illustrate the process of normalizing the data shown in Table 2.20 to the Third Normal form (3NF).
- Identify the primary, alternate, and foreign keys in your 3NF relations.

inserting, updating and deleting data. Some common database management systems include MySQL, PostgreSQL, Microsoft Access, SQL Server, FileMaker, Oracle, RDBMS, dBASE, Clipper, and FoxPro.

### Structured Query Language

Structured Query Language (SQL) is a database programming language designed to retrieve and manage data in a relational database.

### Exercise 2.6

- Describe the steps you would take to convert unnormalised tables to the First Normal Form, Second Normal Form, and Third Normal Form.
- Provide one (1) example relevant to a school or college environment that illustrates reasons for converting database tables to the First, Second and Third Normal Forms.
- How does normalization help in improving the design of a database?
- What kind of inefficiencies may exist in tables that are not normalised?

### Importance of Structured Query Language

The following are the importance of SQL:

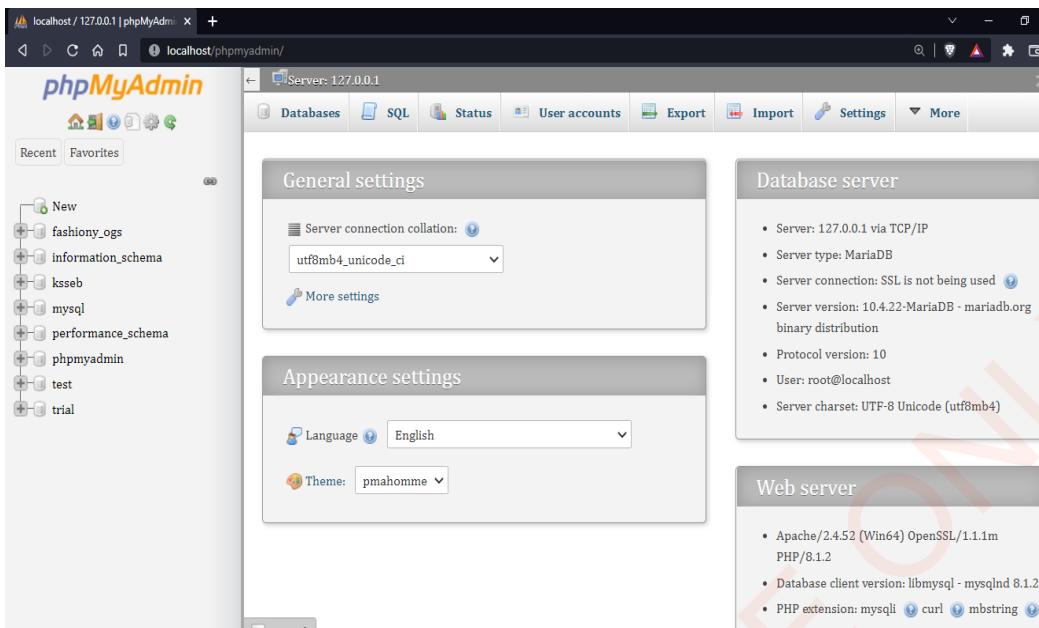
- Allows users to access data in the relational database management systems
- Enables users to describe the data
- Allows users to define the data in a database and manipulate that data.
- Allows embedding within other languages using SQL modules, libraries, and pre-compilers
- Enables users to create and drop databases and tables
- Enables users to create, view, and store procedures and functions in a database
- Allows users to set permissions on tables, procedures, and views

Since there are so many database management systems in this chapter, you will learn about Structured Query Language using MySQL. To use this, you need to install the XAMPP application. Download and install XAMPP, then start MySQL and Apache on the XAMPP control panel to access the database; open your browser and type localhost/phpMyAdmin, and the

### Database management

Database management is the process of organizing, storing and retrieving data from a computer. It involves designing, implementing and sustaining stored data to optimize its value, which database management represents. In designing a system, a programmer must have a database to store data. For example, if you want to store student results in an academic records system, you need to have a database so that all the data can be stored in it. A database enables you to perform many operations, such as

browser will open the dashboard as shown in Figure 2.21.



**Figure 2.21:** *PhpMyAdmin Dashboard*

## SQL commands

There are five commands in SQL language: Data Definition Language (DDL) commands, Data Manipulation Language (DML) commands, Data Manipulation Language (DML) commands, Data Query Language (DQL), Data Control Language (DCL) and Transaction Control Language (TCL). It is crucial to keep the following considerations in mind before diving into SQL:

- SQL statement starts with SQL command and ends with a semicolon (;);
- SQL statements are case-insensitive, which means they can be written using upper, lower, or mixed cases. But it is a common practice to write out SQL keyword commands in all upper cases;

- Using SQL, you can perform most of the actions in a database; and
- SQL data types are binary, numeric, extract numeric, and date.

### Data Definition Language (DDL) commands

DDL commands change the database's structure, for example, by creating a table, deleting a table, and altering a table. DDL has the following commands: CREATE, ALTER, DROP, and TRUNCATE.

#### ***CREATE***

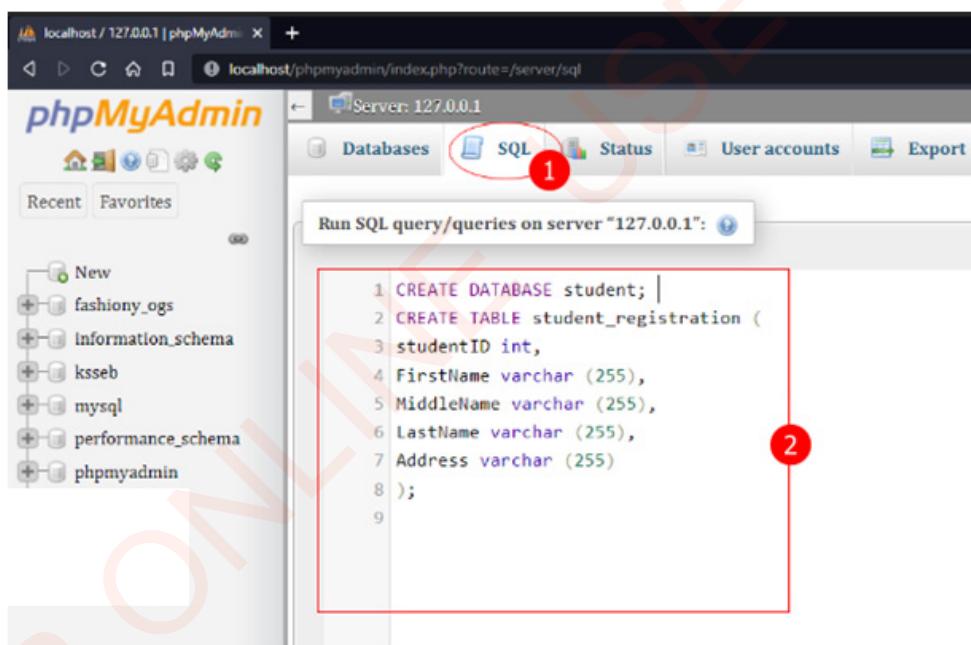
The CREATE command is used to create a new database or table in the database. It has the following syntax: *CREATE DATABASE database\_name*; for the database and *CREATE TABLE table\_name (column\_name datatypes [,])*; for the table.

The database has to be created first then, followed by the tables;

Example:

```
CREATE DATABASE student;
CREATE TABLE student_registration (
    StudentID int,
    FirstName varchar (255),
    MiddleName varchar (255),
    LastName varchar (255),
    Address varchar (255)
);
```

To create a database and table resulting from this example, select SQL to continue with the code and type the code; first, you need to create a database and then the table as indicated in Figure 2.22: this is because the table shall belongs to the database.



**Figure 2.22:** Creating database and table.

When you run the code, it will create the database, and inside the database, the table will also be created, as shown in Figure 2.23.

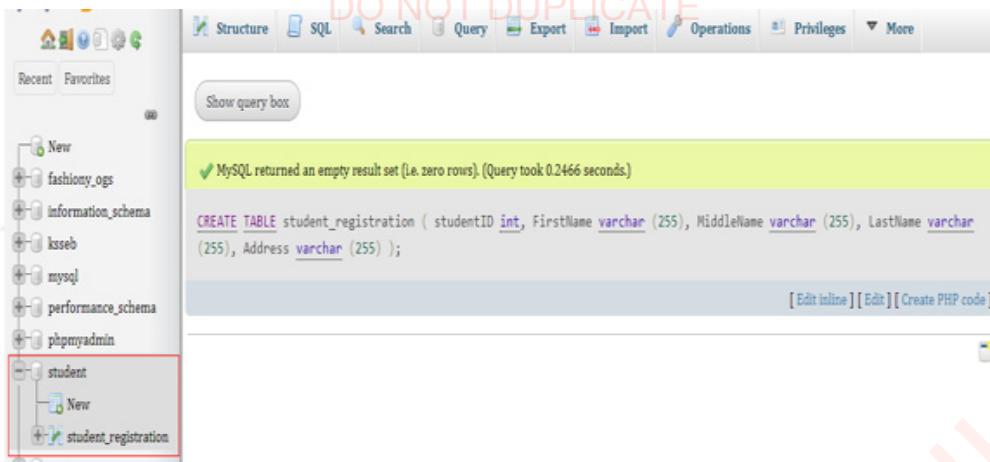


Figure 2.23: SQL query for creating a table

## ALTER

Alter command is a statement used to add, delete, or modify columns in an existing table. The syntax to alter a table by adding a column is *ALTER TABLE table\_name ADD column\_name datatype;* For example, if you want to add a parent name in our database named “student” in the table *student\_registration*, the SQL command will be:

```
ALTER TABLE student_registration
ADD ParentName varchar(255);
```

This will add the *ParentName* column to the *student\_registration* table. To delete a column in a table, use the following syntax *ALTER TABLE table\_name DROP COLUMN column\_name;*

For example: We need to delete *ParentName* from the *student\_registration* table; the command will be:

```
ALTER TABLE student_registration
DROP COLUMN ParentName;
```

To modify a column data type in a table, use the following syntax *ALTER TABLE table\_name MODIFY COLUMN column\_name datatype;*

## DROP

The drop command is used to delete both the table structure and the record stored in the table. The syntax is *DROP TABLE table\_name;*

For example: If you want to delete table *student\_registration*, the command will be: *DROP TABLE student\_registration.* This will delete the table and its contents.

## TRUNCATE

It is used to delete all rows in the table. The syntax is *TRUNCATE TABLE table\_name;*

For example: If you have to delete the rows in the *student\_registration* table, you need to write *TRUNCATE TABLE student\_registration;*

## Data Manipulation Language (DML) commands

DML commands are used to modify the database. Those commands are responsible for all forms of CHANGES in the database. The command of DML is not auto committed, which means it cannot permanently save all the changes in the database. They can be rolled back. DML involves INSERT, UPDATE, and DELETE commands.

### **INSERT**

The insert command is used to add new records to the table. The syntax is *INSERT INTO table\_name (column1, column2, column3, ...) VALUES ('value1', 'value2', 'value3', ...);*

Example: Insert students' data in *student\_registration* table with column (studentID, FirstName, MiddleName, LastName, Address).

### **An example of an INSERT query**

*INSERT INTO student\_registration (StudentID, FirstName, MiddleName, LastName, Address) VALUES('1234', 'Mwakalinga', 'Mwakatope', 'Mkinga', '972 Mbeya');*  
Also, see the query in Figure 2.24.

```

1 INSERT INTO student_registration
2 (studentID, FirstName, MiddleName, LastName, Address)
3 VALUES('1234', 'Mwakalinga', 'Mwakatope', 'Mkinga', '972 Mbeya');

```

**Figure 2.24:** The Insert Query.

If you run this query, it will insert data in the table, as shown in Figure 2.25.

studentID	FirstName	MiddleName	LastName	Address
1234	Mwakalinga	Mwakatope	Mkinga	972 Mbeya

Show all | Number of rows: 25  Filter rows: Search this table

**Figure 2.25:** Inserted data in the table.

### **UPDATE**

The update command is used to update the data value in a table column. After inserting data, you can modify it by using the update command. The syntax is: *UPDATE table\_name SET column1 = 'value1', column2 = 'value2', WHERE condition;*

## FOR ONLINE USE ONLY

*WHERE* is used to specify which data should be modified. If the condition is not specified, ALL the rows in a table will be modified.

For example, suppose you want to update the student's first name with studentID number 1234 from Mwakalinga to Musawomda. In that case, you will apply the following query:

```
UPDATE student_registration
SET FirstName = 'Musawomda'
WHERE studentId = 1234;
```

**Output** is as shown in Figure 2.26 (a) and (b).

```
UPDATE `student_registration` SET `FirstName`='Musawomda' WHERE `studentId`=1234;
```

Figure 2.28 (a): Update query.

studentId	FirstName	MiddleName	LastName	Address
1234	Musawomda	Mwakatope	Mkinga	972 Mbeya

Figure 2.26 (b): Select query of the updated record.

### **DELETE**

The **DELETE** command is used to delete existing records in a table. After inserting data, you can delete the data by using the delete command. The syntax is *DELETE FROM table\_name WHERE condition*:

For example: if you want to delete all the information about Musawomda from our previous example, the query will be: *DELETE FROM student\_registration WHERE FirstName=' Musawomda '*; this query will delete all records with the first name Musawomda. If you want to delete only the record for Musawomda with studentId 1234, you should write the following query:

```
DELETE FROM student_registration
WHERE studentId=1234;
```

### **Data Query Language (DQL) command**

DQL is used to fetch the data from the database. It uses only one command, which is *SELECT*. Its syntax is *SELECT column1, column2, ... FROM table\_name;*

Example: If you want to retrieve the first name and middle name only from the student-registration table, the query is such that:

```
SELECT FirstName, MiddleName
FROM student_registration
```

OUTPUT (see Figure 2.27):

FirstName	MiddleName
Mwakatope	Mkinga
Mkinga	Mwakalinga
Mwakalinga	Mwakatope

Figure 2.27: The output of the *SELECT* query

## Data Control Language (DCL) command

It is used to control users' rights in accessing data in the database. It uses two controls, GRANT and REVOKE.

### **GRANT**

It is used to give users access privileges to a database. The syntax is *GRANT privilege [,privilege], ON privilege\_level TO account\_name;*

For example, If you need to give user\_M access to the database student, the query will be:

```
GRANT ALL
ON student
TO user_M;
```

### **REVOKE**

It is used to revoke access permissions that were previously granted to prevent a user from accessing the database objects. The syntax is *REVOKE privilege\_name ON object\_name FROM {user\_name |PUBLIC |role\_name}*

For example: If you need to remove user\_M from accessing the database student, the query will be:

```
REVOKE ALL
ON student
FROM user_M;
```

## Transaction Control Language (TCL) command

Transaction Control Language command is used to handle database transactions. It keeps track of the modifications made by the data manipulation language. It also enables the grouping of statements

into logical transactions. The transaction control language consists of COMMIT, ROLLBACK, and SAVEPOINT.

### **COMMIT**

COMMIT command is used to save any transaction into the database permanently. When we use any DML command like INSERT, UPDATE or DELETE, the changes made by these commands are not permanent; until the current session is closed, the changes made by these commands can be rolled back. To avoid that, we use the COMMIT command to mark the changes as permanent. The syntax is *COMMIT;*

Example:

```
Delete from student_registration
Where address = 250 Dodoma;
Commit;
```

### **ROLLBACK**

If you have used the UPDATE command to make some changes to the database and realise that those changes were not required, you can use the ROLLBACK command to roll back those changes if they were not committed using the COMMIT command.

Example:

```
Delete from student_registration
Where address = 250 Dodoma;
Rollback;
```

### **SAVEPOINT**

It is used to roll back the transaction to a certain point without rolling back the entire transaction. The syntax for the savepoint commands is *SAVEPOINT savepoint\_name;*

Example:

```
UPDATE class SET name = 'Mwamtumu'
WHERE id = '5';
SAVEPOINT A;
```



### Activity 2.10:

Think of any problem needing a database table with 10 columns, design it, create it, and then populate the table with SQL and fill it.

### Exercise 2.7

1. What is the difference between SQL and MySQL?
2. What do you mean by DBMS? What are its different types?
3. Describe the five groups of SQL commands.
4. What is the difference between DELETE and TRUNCATE commands?

## General database administration

A person who manages the database daily is called a database administrator (DBA).

The primary responsibility of a database administrator (DBA) is to guarantee that data is available, secure from loss and damage, and conveniently accessible when needed. The administrator will plan for whom to use what and when. User role in the database is essential; for example, a super admin accesses all the privileges; this user assigns roles as discussed in Data Control Language (DCL).

## Security features for database

Security and protection are two of the most critical and continually changing aspects of database administration. The major DBMS suppliers have been introducing security features and enhancing how you can protect your valuable corporate data. However, keeping up with these features can be challenging; so, let's take a quick look at some of the more exciting and helpful database security choices.

### Encryption

Even though encryption has been available for a long time, it has only recently become a significant part of database security for securing sensitive data. When data is encrypted, it is altered using an algorithm to be incomprehensible to anyone who does not have the decryption key. The main goal is to make decrypting so tricky that the effort of accessing the data outweighs the gain benefit to a hacker. Data encryption can be used in two scenarios: data in transit and data at rest. Data "at rest" encryption protects data kept in a database, whereas data "in transit" encryption protects data transported over a network.

### Label-based access control

Label-based access control (LBAC) is now available in many DBMSs, providing greater fine-grained control over authorisation to specific data in the database. LBAC can accommodate applications that require a more granular security strategy. LBAC can be configured to control who can read and alter data in specific rows and columns.

**Staying up to date**

Maintaining current knowledge of your DBMS's security requirements and capabilities. Understand what you have at your disposal and what you may need to supplement with extra tools. Also, keep in mind that the security features discussed here are not the only ones available to you for protecting your database data.

**Activity 2.11:**

Assigning roles to users in the MySQL database

Create a database of a law company, create three roles: the super admin, the company manager, and the head of human resources, and assign those roles to Moris, Salima and Berth respectively.

**Chapter summary**

1. Today, people and organisations use various information systems for handling their daily activities socially, economically, and spiritually in areas such as education, healthcare, water, power grid, restaurants, religion, transport, travelling, tourism, and entertainment.
2. In this chapter, you have learned about these information systems and how to create, manage and apply them in real life.
3. The covered areas are an overview of information systems focusing on their elements, components and type and database systems focusing on their advantages, database

models, and common relational database terminologies. Other relational databases emphasise relational database models, types of the relational data model, the Entity-Relationship model, and normalisation.

4. Furthermore, the chapter presented database management focusing on Structured Query Language (SQL) involving DDL, DML, DCL and TCL and the general database administration with a focus on security features to ensure the availability and accessibility of the database.

**Revision exercise**

1. What is an information system? Give examples from your community.
2. Differentiate between any two components of the information system.
3. There are four main types of information systems. With an example, briefly describe how they interrelate.
4. What are the six components that make up an information system? Describe the role of each component.
5. What are the three examples of information system hardware? Identify the role of each of them.
6. What is the most commonly used database model? Differentiate it from the rest.

7. What is normalisation? Explain its importance in relational database design.
8. Keys are an essential part of the relational database. They are used to establish and identify the relation between tables. Using an example, describe various keys applicable in database implementation.
9. What is a functional dependency? State how it relates to, and differs from normalisation.
10. What are some of the most popular Relational Database Management Systems?
11. From the database administration perspective, differentiate the application of encryption and label-based access control and then outline the importance of each.
12. Explain how you can test for NULL values in a SQL database.
13. Identify what is wrong with the below query.  

```
SELECT EmployeeID, AVG(Salary)
FROM EmployeeDetails
WHERE AVG(Salary) > 75
GROUP BY EmployeeID
```
14. Write a SQL statement to create a table named as country that contains columns *country\_id*, *country\_name* and *region\_id*.

15. Given two tables created and populated as follows,
- ```
CREATE TABLE dbo.envelope
(id int, user_id int);

CREATE TABLE dbo.docs(idnum
int, pageseq int, doctext varchar
(100));

INSERT INTO dbo.envelope
VALUES
(1,1),
(2,2),
(3,3);

INSERT INTO dbo.docs
(idnum,pageseq) VALUES
(1,5),
(2,6),
(null,0);
```
- then, what will the result be from the following query? Explain your answer?
- (i) `UPDATE docs SET  
pageseq=7;`
  - (ii) `UPDATE docs SET  
pageseq=11 WHERE  
IDNUM=2;`
  - (iii) `DELETE FROM envelope  
WHERE id=3;`
  - (iv) `UPDATE envelope WHERE  
id=73;`

# Data communication and networking

## Introduction

The transmission of digital data across a network or to a device other than the sending device is the foundation of modern telecommunications. Telephones, computers, Internet, and wireless technology have revolutionised society in how they interact and work due to data communication and networking. Without these, modern civilization would be substantially different. In this chapter you will learn data communication and computer networks, network, topologies and network architecture. The competence develop will enable you to plan and implement a simple data communication network in a school or home environment.

## Concept of data communication

### Data and information

Data refers to the raw facts collected, such as text, images, and symbols, while information refers to the processed data that enables us to take decisions. For example, when the result of a particular test is declared, it contains data from all students. When you find the marks you have scored, you have the information that lets you know whether you have passed or failed. The data transmission is from one point, called the source or transmitter to another point, called a destination or receiver.

Data transmission can be defined as the exchange of data between two devices via some form of transmission media such as a wire cable or wireless. For data communications to occur, the communicating devices must be part of a communication system made up of hardware and software.

### Characteristics of data communication

The effectiveness of any data communication system depends upon the following four fundamental characteristics:

- Delivery: The system must deliver data to the correct destination. Data must be received by the intended device or user and only by that device or user.
- Accuracy: The communication system should deliver the data accurately, without introducing any errors. The data may get corrupted during transmission, affecting the delivered data's accuracy. Data that have been altered in transmission and left uncorrected are unusable.
- Timeliness: Audio and video data have to be delivered on time without delay. Such a data delivery is called real-time data transmission.

- (d) An unsteady: It is the variation in the packet arrival time. Uneven signals may affect the timeliness of data being transmitted.

### Data representation forms

Data is the collection of raw facts which is processed to deduce information. There are different forms in which data can be represented. Some data representation forms used in communications are as follows:

- (a) Text: Text includes a combination of alphabets in small cases and upper cases. It is stored as a pattern of bits. Prevalent encoding systems are American Standard Code for Information Interchange (ASCII) and Unicode
- (b) Numbers: Numbers include a combination of digits from 0 to 9. It is stored as a pattern of bits. Prevalent encoding system are ASCII and Unicode
- (c) Images: Is a visual representation of something. In computers, images are digitally stored. It is formed by picture elements called a pixel. A pixel is found in the image and it is the smallest element of an image. In simple terms, a picture or image is a *matrix of pixel elements*.

The pixels are represented in the form of bits whereby each pixel would require a different number of bits to represent the value of a pixel depending on the type of image (black and white or coloured).

The size of an image depends on the number of pixels (also called

resolution) and the bit pattern used to indicate the value of each pixel.

*Example:* If an image is purely black and white (two colours), each pixel can be represented by a value of 0 or 1. So, an image made up of  $10 \times 10$ -pixel elements would require only 100 bits in memory to be stored.

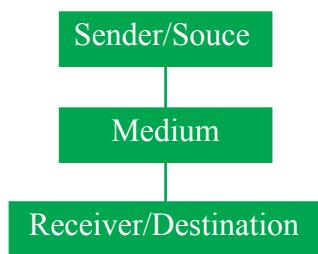
- (d) Audio: Data can also be in the form of sound, which can be recorded and broadcast. For example what we hear on the radio is a source of data or information. Audio data is continuous, not discrete.
- (e) Video: Video refers to the broadcasting of data in the form of moving pictures or movies.

### Data communication model

Generally, the data communication model represents the holistic structure of the communication system.

### Basic components of the data communication model

A basic data communication model consists of three components: source, medium, and destination. Figure 3.1 shows the components of the basic data communication model.



**Figure 3.1:** Components of data communication

## The description of the communication model elements

**Sender/Source:** The device that generates the data transmission is known as the sender, for example, a terminal computer or mainframe computer.

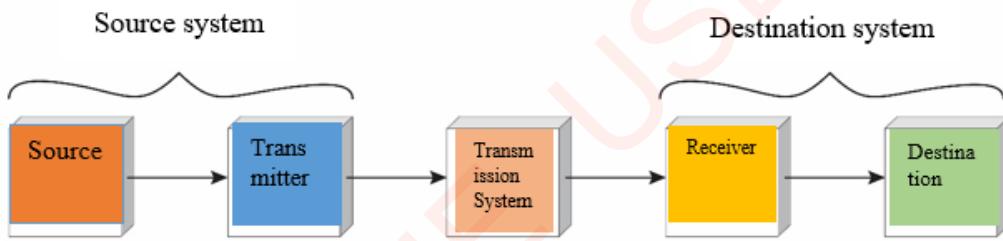
**Transmission medium:** The path through which the message passes or propagates from the sender to the recipient—for example, cables, microwaves, fibre optics, and radio frequency.

**Receiver:** The device that receives the message, for example, printer, terminal, computer, and mainframe.

## Data communication models key elements

The exchange of data between two parties is the primary goal of a communications system.

Let us look at a simple communication paradigm whereby the significant features of the model were invented by Shannon (1948). According to Shannon, data communication can be modelled using a systematic view with five components. These components are the source, transmitter, medium, receiver, and destination, as shown in Figure 3.2.



**Figure 3.2: Data communication model block diagram**

## The description of each data communication model component

**Source:** The data to be transferred is generated by this device; examples include telephones and personal computers.

**Transmitter:** The data generated by a source system is not directly delivered in the form it was generated. A transmitter converts and encodes that data into a format appropriate for transmission before transmitting it via a transmission system.

**Transmission System:** A transmission system is a single transmission line or medium that connects the source and destination and allows them to transfer data.

**Receiver:** The receiver receives and transforms the signal from the transmission system.

**Destination:** This is the point that receives data from the sender.

### Data communication modes

The process of sending data between two devices connected over a network is a communication mode or method. These modes control the direction of information flow.

Assume that you have two devices that need to communicate, whereby device A is a transmitting device and device B is a receiving device; TxD in data communication represents the transmitting device, and RxD represents the receiving device. Data transferred over the medium can either use analog or digital communication systems.

*Analog communication mode* is a transmission mode made of the continuous varying waveforms that are transmitted between TxD and RxD, as shown in Figure 3.3(a).

### Advantages of analog communication

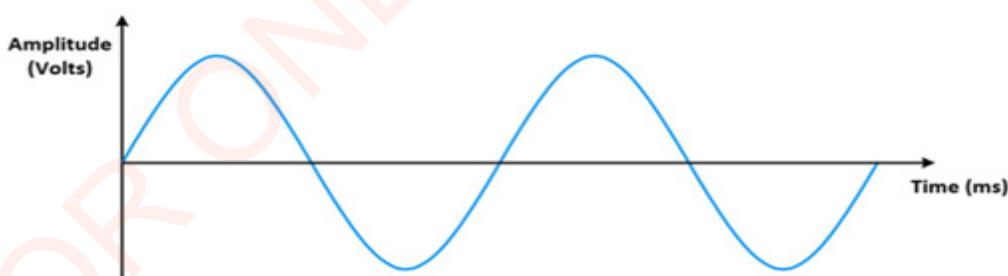
Analog communication offers the following advantages:

- They use less bandwidth
- They are easier to process
- They are inexpensive

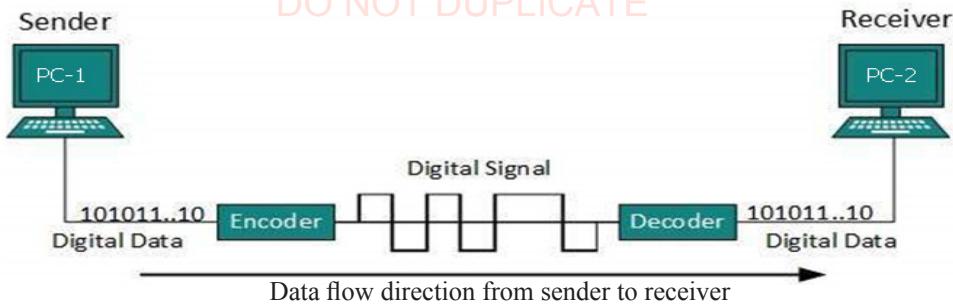
### Disadvantages of analog communication

- Analog signals usually are lower quality signals than digital signals.
- Data transmission at long distances may result in undesirable signal disturbances.
- Analog signals are prone to generation loss.
- Analog signals are subject to noise and distortion compared to digital signals with much higher immunity.

**Digital communication mode** transfers the message signal in digital form, that is to say, in a non-continuous discrete form, as shown in Fig 3.3(b)



**Figure 3.3: (a) Analog data signal**



**Figure 3.3(b): Digital data signal**

### Advantages of digital communication

- (a) In digital communication, the speech, video and other data may be merged and transmitted over a shared channel using the multiplexing technique. Through multiplexing techniques, digital communication can combine multiple signals into one shared medium.
- (b) The digital communication system is simpler and cheaper because of the advancement of integrated circuit (IC) technologies.
- (c) It has a mechanism for controlling the amount of interference.
- (d) The devices used are mostly similar in digital communication because they have a common structure for encoding signals.
- (e) The digital communication system has more protection against noise and outdoor interference.
- (f) Digital communications provide added security to information signals because the transmission of information is done in digital or encoded form, which means the signals are encrypted.

- (g) Digital information can be saved and retrieved when required, while it is not possible in analog.
- (h) The error correction and detection techniques are implemented efficiently in the digital communication system.
- (i) Digital communication is cheaper compared to analog communication.

It is simpler to design and configure digital communication systems than analog ones, even though they are challenging.

### Disadvantages of digital communication

- (a) Due to analog to digital conversion, the data rate becomes high; therefore, more transmission bandwidth is required for digital communication.
- (b) Synchronization is required in digital communication during the process of synchronous modulation.
- (c) Digital communication has high power consumption rate.



### Activity 3.1: Advantages and disadvantages of analog and digital television

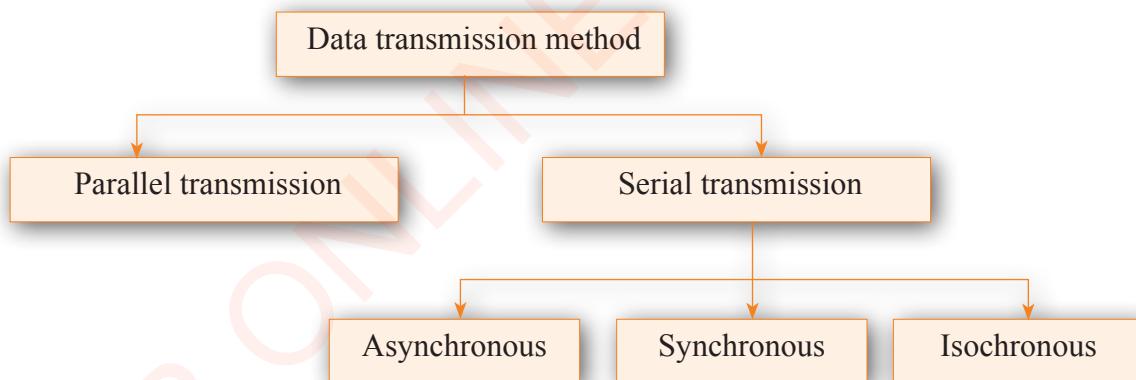
In 2013, the Government of Tanzania was the first East African country to migrate from analogue broadcasting to digital terrestrial television broadcasting. It directed analog broadcasting companies to switch off their devices. In your own words, discuss both the advantages and disadvantages of analog and digital television broadcasting.

To ensure that each device can send to and receive data from other digital, the International Standard Organization (ISO) established interface standards to govern hardware designers not to create their own signalling representation for their devices; hence harmonizing the intercommunication between different hardware devices regardless of vendor.

### Classification of data transmission mode

Data can be transmitted from source to destination device in different ways. This section describes various classifications of data transmission modes based on three criteria, as shown in Figure 3.4.

- (a) Based on the number of wire connections and number of bits sent simultaneously:
  - (i) Parallel transmission
  - (ii) Serial transmission
- (b) Based on bit synchronization and time control:
  - (i) Asynchronous
  - (ii) Synchronous
- (c) Based on the direction of signal flow or direction of exchange of the information:
  - (i) Simplex
  - (ii) Half-duplex
  - (iii) Full-duplex



**Figure 3.4:** Types of data transmission modes

*Based on the number of wire connections and the number of bits sent simultaneously*

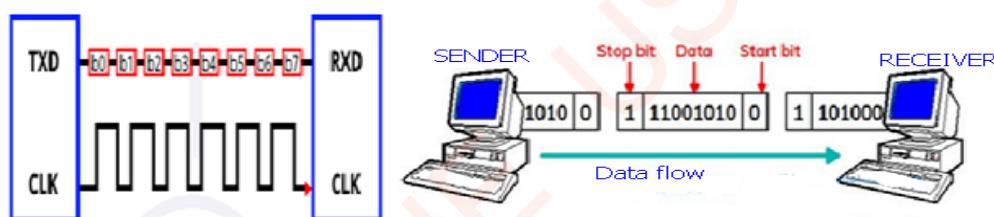
### **Serial transmission**

Sending data sequentially across a data transmission medium is called serial communication. In serial transmission, data are transmitted bit after bit. The Least Significant Bit (LSB) is usually transmitted first. While sending data serially, characters or bytes are broken into bits before being transmitted. As a result, hardware is needed to transform the data from parallel to serial. At the destination, all of the bits are gathered, measured, and assembled as bytes in the destination's memory. This necessitates a serial-to-parallel conversion.

As opposed to parallel transmission, the

serial transmission uses only one circuit to connect the two devices. As a result, serial transmission is appropriate for long-distance transmission, as shown in Figure 3.5.

Data are usually transmitted as packets. Each packet is given a packet identifier (packet ID) before being transmitted for the proper reception and reassembling of data on the receiver side. The packet ID is the part of a small header that is added at the beginning of each packet of data created as a result of the fragmentation of the original message. The header of each packet contains other information like routing information (source and destination IP address and Media Access Control (MAC) address).



**Figure 3.5: Serial data communication**

### **Serial transmission is used when**

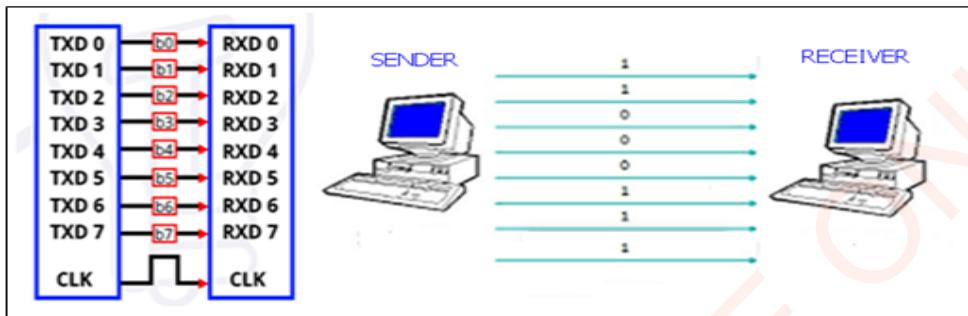
- The data is to be transmitted over a long distance;
- In cases where the amount of data being sent is relatively small.

### **Parallel transmission**

As demonstrated in Figure 3.6, the parallel transmission simultaneously sends all of the bits of a byte on separate channels. Multiple connections between the two devices are thus required in this case. This is the fastest means of sending data from one location to another. The disadvantage of parallel transmission is that, it is quite expensive because it necessitates using many lines for both transmitting and receiving devices. Second, it necessitates remarkable precision, which cannot be guaranteed across large distances.

- Parallel transmission is used when:
- A large amount of digital data is to be transmitted
  - The digital data to be sent is time-sensitive; and
  - The digital data needs to be sent quickly.

For example, parallel transmission is used to send data in video streaming. Since video streaming requires the transmission of large volumes of data, the data being sent is also time-sensitive as slow data streams result in poor viewer experience. Figure 3.6 shows parallel data communication.



**Figure 3.6: Parallel data communication**

Table 3.1 shows the difference between serial and parallel transmission.

**Table 3. 1: Comparison between serial and parallel transmission**

| Serial Transmission                                     | Parallel Transmission                                    |
|---------------------------------------------------------|----------------------------------------------------------|
| Data flows in both direction bit by bit                 | Data flows in multiple directions, 8 bits at a time      |
| Inexpensive                                             | Expensive                                                |
| The number of bits transferred per clock pulse is 1 bit | The number of bits transmitted per clock pulse is 8 bits |
| Used for low-speed data transmission                    | Used for high-speed data transmission                    |
| Used for long-distance communication                    | Used for short-distance communication                    |
| It can be used to connect computer to computer          | It can be used to connect the computer to the printer    |

#### *Based on bit synchronization and time control*

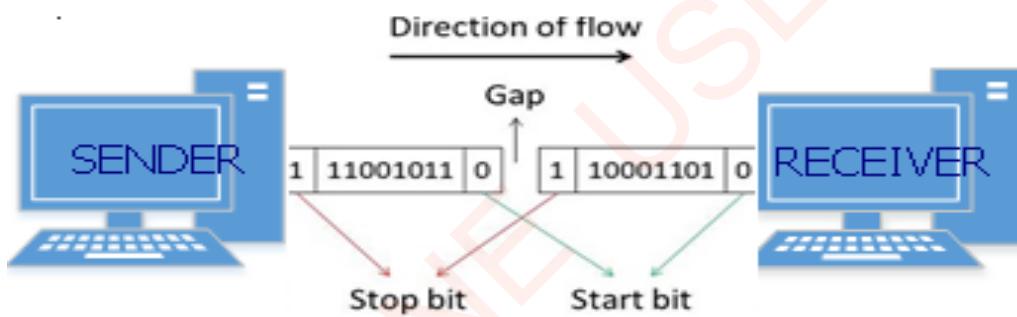
One of the most difficult aspects of data transmission is keeping the receiver (destination) synchronised with the sender (source). This is the core concern with serial communication. The receiver must be able to recognize the beginning of each new character in the bit stream that is being presented to it; otherwise, it will be unable to interpret the incoming bit stream correctly.

There are two main types of serial data transmission methods based on bit synchronization and time control. These are:

- (a) Asynchronous
- (b) Synchronous

These types of transmission use bit synchronization. Bit synchronization is necessary to identify the beginning and end of the data transmission. Bit synchronization supports the receiving computer in recognising when data transmission begins and ends. Therefore, bit synchronization offers timing control.

**Asynchronous communication:** This sends one character at a time, separated by a start bit and one or two stop bits as in Figure 3.7. Every frame starts with a start bit, which allows the receiving device to adjust to the timing of the sent signal. The message can start whenever it wants. Messages are kept as short as possible in this case because the sending and receiving devices should not be out of sync while the message is being sent. Asynchronous communication is the most often used method for transmitting character data and is best suited for characters that are delivered at irregular intervals, such as when users type in character data from the keyboard.



**Figure 3.7: Asynchronous communication mode**

A frame used to transmit character data has four components:

**A start bit:** It is starting frame and enables the receiving device to synchronize itself with the message. It tells the receiver the end of one bit in the data transmission process.

**Data Bits:** Consists of 7 or 8 bits when character data is being transmitted.

**Parity Bits:** Eight bits are transmitted at a time with a start and stop bit which are 0 and 1, respectively. These bits (start and stop) are known as Parity bits are used as a crude method for detecting transmission errors.

**The parity bit:** Is used for error detection in asynchronous transmission. Parity approaches can detect errors that affect only one bit; but, if errors affect two or more bits, parity techniques may be unable to detect them.

**A stop bit or bits:** Signals the end of the data frame.

### Advantages of asynchronous communication

- Asynchronous transmission is easy and affordable.
- It is perfect for transferring small frames at irregular intervals (e.g., data entry from a keyboard). Because each individual character is complete in itself, if a character is corrupted during transmission, its successor and predecessor are unaffected.
- No synchronization is required between the transmitter and receiver devices. The sender can directly transmit data and the receiver can receive that data.
- The sender does not require any primary storage device.
- The cost is very low to implement this method.

### Disadvantages of asynchronous communication

- Unsuitable for huge amounts of data transmission.
- Less efficient.

### The efficiency of asynchronous transmission:

In order to evaluate the efficiency of the asynchronous data transmission, the following formula is applicable:

Where,

- Actual data bits refer to the amount of data bits to be sent over the medium.
- Total bits refer to the sum of actual data bits and overhead data bits.

(c) Overhead data bits are start bit (1 bit), stop bit (1 bit), and parity bit (1 bit).

**Example 1:** Determine the efficiency of 30KB data transmission using the Asynchronous transmission method.

#### Solution:

$$\text{Actual data} = 30\text{KB} = 30 \times 8\text{Kb} = 240\text{Kb}$$

$$= 1000 \times 240\text{b} = 240,000 \text{ bit}$$

Overhead data bit required for each group of 8 bit = 3 bit

So, total overhead data bits required for 240000 bit data transmission

$$= (3/8) \times 240,000 = 90,000 \text{ bit}$$

$$\text{Total bits} = \text{Actual data bit} + \text{overhead data bits} = 240,000 \text{ bit} + 90,000 \text{ bit}$$

$$= 330,000 \text{ bit}$$

So, efficiency

$$= (240,000/330,000) \times 100\% = 72\%$$

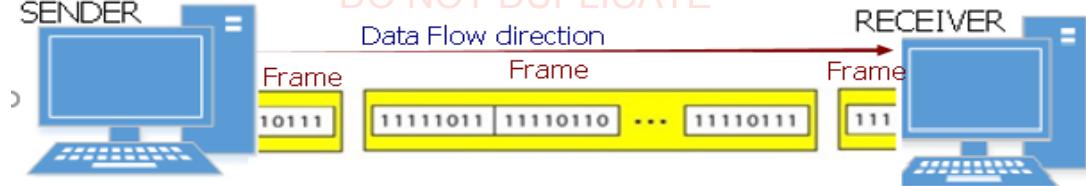
### Synchronous communication

Instead of sending one character at a time, synchronous communication sends an entire block (chunks or frames) of data bits at once. Transmission starts at a predefined regular time interval in this case. A synchronisation signal is used to inform the receiving station of the arrival of a new frame and to synchronize the receiving station as illustrated in Figure 3.8.

In this method, bit stream is combined into longer frames that may contain multiple bytes.

**Note:** Data is stored in memory before it can be transmitted to divide it into several blocks. Then, it is transmitted block by block with regular interval. A block consists of several characters (normally 128, 256, 512, 1024 characters) with header information (16 bits) at the beginning and trailer information (16 bits) at the end.

FOR ONLINE USE ONLY  
DO NOT DUPLICATE



**Figure 3.8:** Synchronous communication

Synchronous signals often use a bit pattern that cannot be found anywhere else in the transmission, ensuring that they are always distinct and easily recognised by the receiver. Frames can be longer since the transmitter and receiver remain in synchronization for the whole period of the transmission.

As frames become longer, the parity method of error detection becomes ineffective because, if many bits are affected, the parity technique will not reliably report the fault. As a result, the Cyclic Redundancy Check (CRC) is employed with the synchronous transmission.

A CRC value is calculated by the transmitter using an algorithm that sums the value of all data bits. The data frame

receives this CRC value appended to it. The receiver does the same algorithm to recalculate the CRC and compare the CRC in the frame to the calculated value. If these values match, the frame was sent successfully.

An end-bit pattern indicates the end of the frame. End's bit pattern is similar to synchronisation signal, in that it does not exist elsewhere in the messages, ensuring that they are always distinct and easily recognised at the end of the frame. For high-speed communication between computers, serial synchronous transmission is used. When large amounts of data must be sent, it is employed. Table 3.2 shows the advantages and disadvantages of synchronous and communication.

**Table 3.2:** Advantages and disadvantages of synchronous communication

| Advantages                                              | Disadvantages                                                                                                                                                                                             |
|---------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Efficiency is comparatively very high.                  | It is comparatively expensive since it requires a complex circuit and its implementation is difficult.                                                                                                    |
| The speed of data transmission is very high.            | The entire data block is lost when a transmission error occurs, not just a single character.                                                                                                              |
| No need to transmit the start and stop bit.             | The sender cannot just transmit characters as they occur but must store them until a block has been built up. As a result, it becomes ineffective, where characters are generated at irregular intervals. |
| It is suitable for a large amount of data transmission. | The primary storage device is required.                                                                                                                                                                   |
|                                                         | Synchronization between the source and target is required.                                                                                                                                                |

***The efficiency of synchronous transmission PLICATE***

In order to evaluate the efficiency of the synchronous data transmission, the following formula is applicable:

$$\text{Efficiency} = \frac{\text{Actual data distance}}{\text{Total bits}} \times 100$$

Where,

- (i) Actual data bits refer to the amount of data bits to be sent over the medium
- (ii) Total bits refer to the sum of actual data bits and overhead data bits
- (iii) Overhead are header information (16 bits) and trailer information (16 bits)

**Example 2:** Determine the efficiency of 30KB data transmission using the asynchronous transmission method.

**Solution:**

$$\text{Actual data} = 30\text{KB} = 30 \times 8\text{Kb} = 240\text{Kb} = 1000 \times 240\text{b} = 240,000 \text{ bit}$$

Suppose, a block has 80 characters, So, block size =  $80 \times 8 \text{ bit} = 640 \text{ bit}$

Overhead data bit required for each block of 640 bit group = 32 bit

$$\begin{aligned} \text{So, total overhead data bits required for } 240,000 \text{ bit data transmission} \\ = 32/640 \times 240,000 = 12,000 \text{ bit} \end{aligned}$$

$$\begin{aligned} \text{Total bits} &= \text{Actual data bit} + \text{overhead data bits} = 240,000 \text{ bit} + 12,000 \text{ bit} \\ &= 252,000 \text{ bit} \end{aligned}$$

$$\text{So, Efficiency} = (240,000/252,000) \times 100\% = 95\%$$

**Note:** The more a block has characters, the more method has efficiency.

The difference between synchronous and asynchronous transmission is as indicated in the Table 3.3.

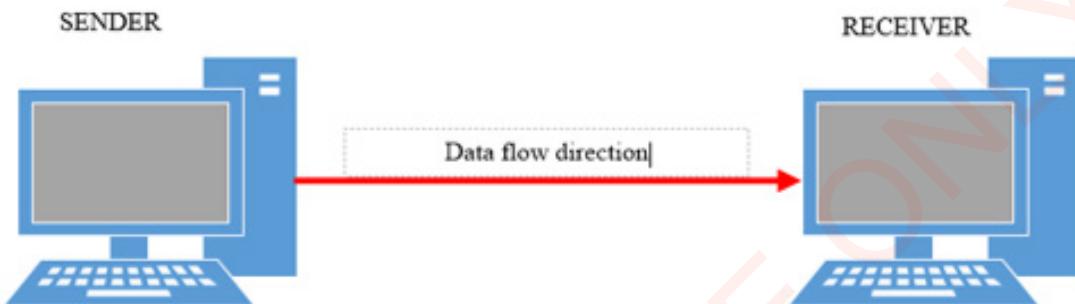
**Table 3.3: Synchronous vs Asynchronous transmission**

| Synchronous Transmission                                                    | Asynchronous Transmission                    |
|-----------------------------------------------------------------------------|----------------------------------------------|
| Transmit data in the form of blocks, chunks or frames                       | Transmits 1 byte or character at a time      |
| The transmission speed is high                                              | The transmission speed is low                |
| Expensive                                                                   | Economical                                   |
| It has a constant time interval                                             | Has random time interval                     |
| There are gaps between the data                                             | There are no gaps between the data           |
| They are used in Chatrooms, Telephone conversations and video conferencing. | They are used in Emails, forums and letters. |

*Based on the direction of signal flow or direction of exchange of the information*

There are three types of transmission modes under this subclass. These are simplex mode, half duplex and full duplex.

**Simplex mode:** Data can only be transmitted in one direction; hence communication is unidirectional. As a result, they cannot convey a message to the sender. Simplex systems use unidirectional communication, in which we only need to provide a command/signal and do not require a response. Loudspeakers, television and remote control, radio call, keyboard and monitor are examples of simplex mode, as in Figure 3.9.



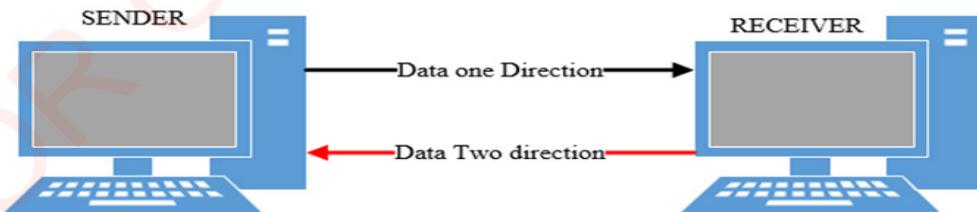
**Figure 3.9: Simplex mode**

### **Half-duplex mode**

Half-duplex data transmission implies that data can be delivered in both directions on a single signal carrier, but not simultaneously.

For example, on a local area network using half-duplex transmission technology, one workstation can send data on the line and then immediately receive data on the line from the same direction in which the data was just transferred. On contrary to the half-duplex transmission, which requires a bidirectional line (one that can transmit data in both ways), data can only be delivered in one direction at a time.

A radio call used by police is an example of half-duplex communication in which messages are sent one at a time but in both directions. Figure 3.10 shows the half-duplex mode.

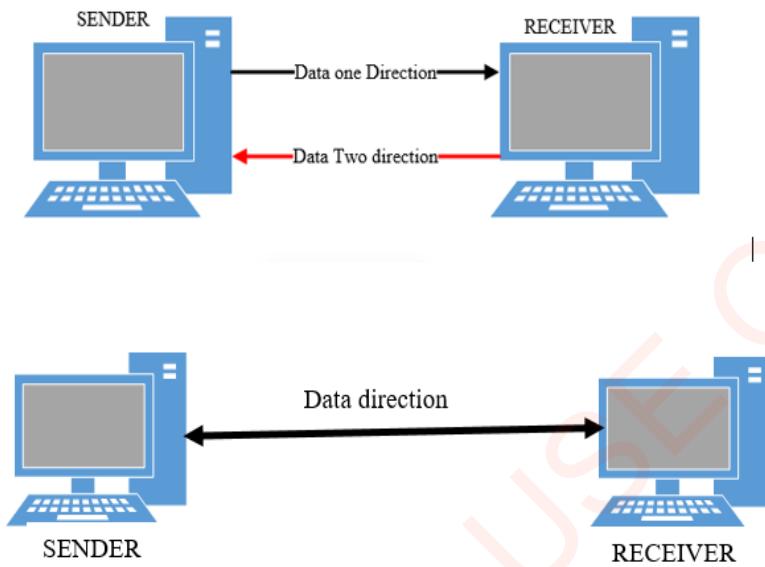


**Figure 3.10: Half-duplex mode**

**Full duplex mode**

Full duplex mode allows data transfer in both directions simultaneously. A telephone network is an example of full duplex communication since it allows two people to talk and listen simultaneously across a telephone line.

There are two lines in a full duplex system, one for sending and receiving data. Figure 3.11 illustrates full-duplex mode.



**Figure 3.11: Full-duplex mode**

**Exercise 3.1**

1. Colour image uses 16 bits to represent a pixel. What is the maximum number of different colours that can be represented?
2. Discuss the concept of data communication system.
3. Calculate the efficiency of 1MB data transmission using asynchronous transmission method.
4. Calculate the efficiency of 1MB data transmission using synchronous transmission method.
5. Discuss the real life application of Simplex, Half-duplex, Full-duplex communication modes at your environment.
6. What are the advantages of digital data over analog data?

## Computer networks DO NOT DUPLICATE

Computer networking has changed the way people communicate information globally and locally. It has also changed business operations, social services, and knowledge sharing. It is on these grounds the study of computer networks becomes essential.

### Concept of computer network

A network is a collection of devices (nodes) linked together by a communication medium. A node can be a computer, printer, or any other device that can send and/or receive data generated by other network nodes.

The computer network can be defined as the collection of computers and other communication devices linked together using communication media for the purpose of data communication and resource sharing. Electronic devices on the network may be connected by cables or wireless.

### Components of computer network

The computer networks environment is made up of the following:

- Network devices include Network Interface Cards (NIC), hubs, switches, routers, repeaters, bridges and modems.
- Network software such as Network Operating Systems (NOS) such as Microsoft Window server and Ubuntu Server. Network software manages end-user devices such as PC, printers and servers.
- Transmission media include bounded transmission media

like Coaxial cable, Twisted and Untwisted pair cable and fiber optics cable; unguided transmission media such as radio waves, microwaves and infrared.

### Classification of computer networks

A computer network can be classified based on various criteria. However, the most common criteria are:

- Classification based on geographical area
- Classification based on ownership/ access restriction
- Classification based on topology
- Classification based on architecture

### *Classification of computer networks based on geographical area*

The geographic area and number of computers that are part of a network should be used to represent the size of a network. It ranges from devices in a single room to millions of devices dispersed worldwide. The following are the most common types of computer networks under this category:

#### **Personal Area Network**

*Personal Area Network (PAN)* is a computer network built around a single person. It comprises a computer, mobile device, or personal digital assistant. PAN connects personal gadgets to a digital network and the internet by establishing communication between them. PAN is the smallest network which is very personal. This type of network normally ranges from around 10 meters. Figure 3.12 shows an example of a Personal Area Network.

**Figure 3.12: Personal Area Network**

Personal Area Networks may include Bluetooth-enabled devices or infrared-enabled devices. When we connect two or more devices using Bluetooth or infrared for data transfer, we create a network in our personal space.

All Bluetooth-enabled devices, like Bluetooth-enabled headphones and speakers can be a part of the Personal Area Network. Also, Infrared enabled devices like TV remotes, cordless mouse/keyboards, and infrared touch screens are also part of the personal area network.

#### **Characteristics of Personal Area Network**

- It is mostly personal devices network equipped within a limited area (approximately 10 meters).
- Allows you to handle the interconnection of digital devices in the surrounding area of a single user.

- It can be wirelessly connected to the internet called WPAN.
- Appliances used for PAN: cordless mice, keyboards, and Bluetooth systems.

#### *Advantages of PAN*

- PAN networks are relatively secure and safe.
- No wire connectivity is required
- Easy data synchronization

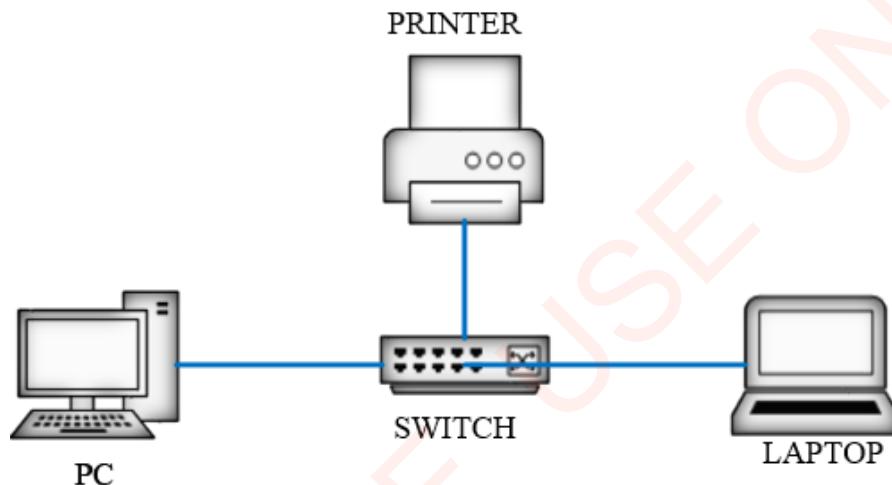
#### *Disadvantages of PAN*

- It may establish a malicious connection to other networks on the same radio channel.
- Bluetooth and infrared have a slow data transfer rate compared to other networks like LAN (local area network).

- (c) Distance limitation: The signal range is a maximum of 10 meters, limiting long-distance sharing.
- (d) A personal area network also uses infrared to interfere with radio signals, and data can be dropped.

### **Local Area Network**

A *Local Area Network (LAN)* is a collection of computer and peripheral devices that are linked in a specific location, such as a school, laboratory, home, or business building. It is a network that is extensively used for sharing resources such as files, printers, games, and other applications. For example, connecting computers and printers at home or office, as shown in Figure 3.13.



**Figure 3.13: Local Area Network**

LAN is a privately owned network, which means anyone can create it. You just need cables like Ethernet cables or central devices like a hub or switch to create it. The Local Area Network is so secure because there is no external connection with the local area network; so the data shared within the LAN network is safe and cannot be accessed outside unless one of the member nodes of the network is connected to the internet. LAN networks are small size networks, so they are faster. Data transfer speed over the LAN network

can reach up to 1000Mbps (1Gbps).

#### *Characteristics of LAN*

The following are the important characteristics of a LAN network:

- (a) It is a private network; so an outside regulatory body has no control over it.
- (b) LAN operates at a relatively higher speed compared to other types of networks.
- (c) Uses various kinds of media access control methods like token ring and Ethernet.

*Disadvantages of LAN*

- LAN application area*
- LAN networks are used in businesses where all business data are stored on servers.
  - LAN can be used in factories.
  - It can be used at schools and colleges where students, teachers and other staff data are stored on the servers.
  - It can also be used in our homes where all computers, mobiles, and printers are connected through central devices (hub/ switch/router) for the aim of data exchange.

*Advantages of LAN*

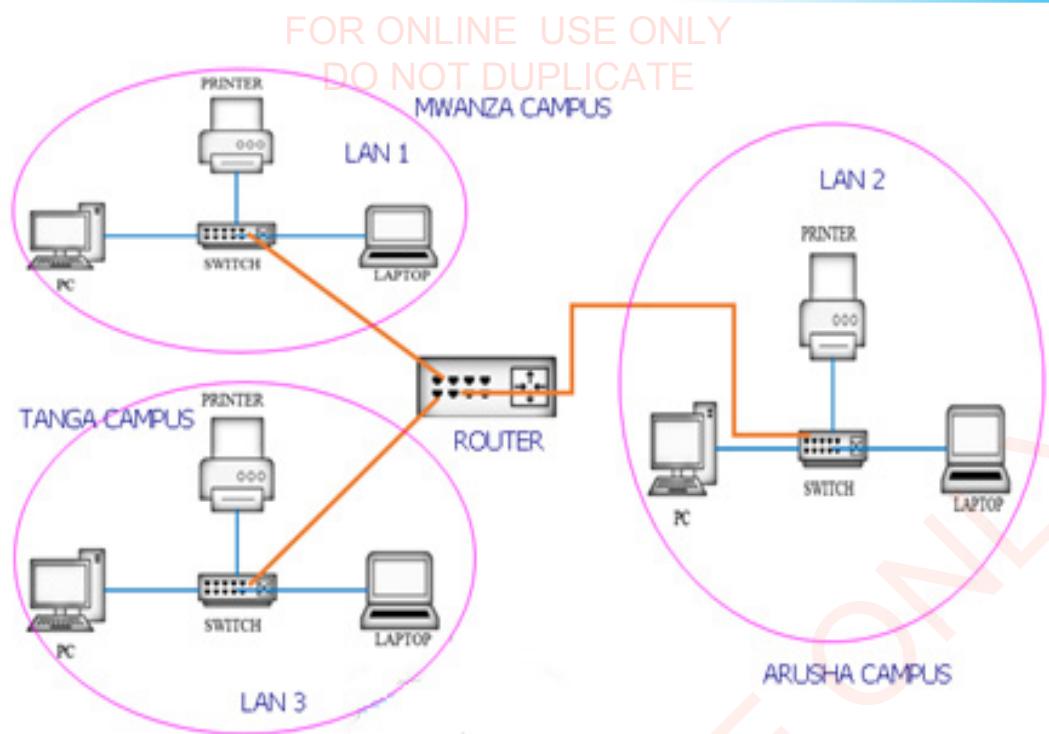
- Local area networks can be used to share computer resources such as hard disks, DVD-ROMs, and printers. This drastically reduces the cost of purchasing hardware.
- Instead of purchasing licensed software for each client in the network, you can use network licenced software over the network which reduces cost.
- Data of all network users can be stored on a single hard disk of the server computer.
- You can quickly send files and messages through networked computers.
- It will be easier to manage data in a single location, making it more secure.
- The Local Area Network allows all LAN users to share a single internet connection.

- Disadvantages of LAN*
- Although LAN saves money due to pooled computer resources, the initial cost of setting up a Local Area Network is relatively high.
  - Because the LAN administrator can see any LAN user's data files, it does not provide adequate privacy.
  - Unauthorized users can gain access to an organization's essential data if the LAN administrator cannot secure the centralized data repository.
  - There are challenges with software configuration and hardware failures; So, Local Area Networks necessitate continual administration.

*Metropolitan Area Network*

Metropolitan Area Network (MAN) is a computer network that extends across an entire city, college campus, or a small region. This network type is larger than LAN, which is mainly limited to a single building or site. Depending upon the type of configuration, this type of network allows you to cover an area from several miles to tens of miles.

In other words, when two or more LANs network are connected for communication, it becomes a MAN network. Example: If the university has branches in different locations and every branch uses a LANs network, as illustrated in Figure 3.14 this types of network is known as MAN. The company connects all LAN networks through a telephone line so that all branches can share data and resources. This type of network will be called MAN.



**Figure 3.14: Metropolitan Area Network**

#### *Characteristics of MAN*

- It covers towns and cities in a maximum 50 km range.
- The most used medium is optical fibre cables.
- It provides an adequate data transmission rate for distributed computing applications.

#### *Applications of MAN*

- Since MAN covers the entire city, this type of network is suitable for cable Television networks.
- It can be used in private industries
- It can be used in universities and colleges
- It can be used in railways and airlines
- It can be used in military areas for communication

#### *Advantages of MAN*

- It offers fast communication using high-speed carriers, like fibre optic cables.
- It provides excellent support for an extensive network and greater WAN access.
- The dual bus in the MAN network supports transmitting data in both directions concurrently.
- A MAN network mostly includes some areas of a city or an entire city.

#### *Disadvantages of MAN*

- Requires more cable to establish MAN connection from one place to another.
- In a MAN network, securing the system from hackers is difficult.

**Wide Area Network**

Wide Area Network (WAN) is a type of computer network that extends over large geographical areas, such as whole country, states, and the continent or entire globe. An example of a Wide Area Network is the internet. When many LANs and MAN networks are connected to each other for the purpose of communication, the formed network will be termed WAN. There are two types of WAN: Switched WAN and Point-to-Point WAN. An example of a Switched WAN is the Asynchronous Transfer Mode (ATM) network, and Point-to-Point WAN is a dial-up line that connects a home computer to the Internet.

*Characteristics of WAN*

- (a) A communication medium used for WAN is Public Switched Telephone Network (PSTN) or Satellite.
- (b) WAN's data rate is slow, about a 10th LAN's speed, since it involves increased distance and an increased number of servers and terminals.
- (c) Software files will be shared among all the users on the network; therefore, all can access the latest files.
- (d) Any organization can form its global integrated network using WAN.
- (e) Optic wires, Microwaves, and Satellites are used to transmit data through WAN.
- (f) Speeds of WAN range from a few kilobits per second (Kbps) to megabits per second (Mbps).

*Advantages of WAN*

- (a) The WAN network is the solution for business offices situated at

longer distances. WAN provides the means by which all offices can be organized and managed easily through this type of network.

- (b) Similar to MAN, the fault tolerance of a WAN is less, and there is more congestion in the network.

*Disadvantages of WAN*

- (a) The initial setup cost of investment is very high.
- (b) Due to long-distance transmission, noise and error are more frequent in WAN.
- (c) Propagation delay is one of the biggest problems faced here in MAN.
- (d) Designing and maintaining the WAN network is difficult; you need skilled technicians and network administrators.
- (e) It requires more time to resolve issues because of the involvement of multiple wired and wireless technologies.
- (f) Offers lower security compared to other types of networks in the computer.

*Network type based on ownership/access restriction*

This type of network is sub-categorised into a public and private networks. A public network is a type of network wherein anyone, namely the general public, has access and can connect to other networks or the Internet through it. This is in contrast to a private network, where restrictions and access rules are established in order to relegate access to a select few.

**Characteristics of Private Network**

- This type of network is owned and controlled by a person or private organization.
- This type of network is not open to all members of the given network. Only registered members can use this network.
- The security of this type of network is extremely strong.
- Has low traffic.
- Computers on this network are not visible to the public.
- The administering authority determines access restrictions.

For example: Different banks have their own network system or Intranet of an organisation.

**Characteristics of public Network**

- This network belongs to a company or Government but allows public service access, usually at a charge.
- This network is not owned and controlled by any person or private organisation, but an organization runs it.
- Computers on this network are visible to the public.

For example, the Internet is a public network because it uses public IP addressing space to enable anyone access to the resources on the internet, mobile phone or telephone network systems etc.

**Note:** Computer networks are also classified based on architecture and topology as will be described in the next subtopics.

**Advantages of computer network**

Computer systems and peripherals are connected to form a network. They provide

numerous advantages, such as:

- Resource sharing like printers and storage devices
- Exchange of information using E-Mails and FTP
- Information sharing by using the Web or the Internet
- Interaction with other users using dynamic web pages

**Network topologies****Meaning of network topology**

Computer network topology is the way various components of a network (like nodes, links and peripherals) are arranged. Network topologies define the layout, virtual shape or structure of the network, not only physically but also logically. How different systems and nodes are connected and communicate with each other is determined by the topology of that given network.

**Classification of network topologies**

Network topologies are categorised into two types which are logical topology and physical topology.

**Logical topology** refers to how data is transmitted from one device to another in the network. There are two types of logical topologies, Ethernet and token ring.

- Ethernet topology:** All computers listen to the network media and can only send data when none of the others is sending.

- (b) *Token ring topology*: In token ring topology, a special package for data called a token goes around the network. This token will be taken by the computers whose addresses were attached to the token. Computers will release the token after reading the data. In contrast to its design, logical topology relates to how data moves through a network.

**Physical topology:** The physical topology refers to the physical layout of a network which includes the devices, installation, and location.

The physical topology can be classified as bus, ring, star, tree, mesh and hybrid. They are complex networks which can be built of two or more topologies. These topologies are described in this section as follows:

**Bus topology:** The usage of a multi-point medium distinguishes bus topology

from others. A backbone is a single long cable connecting all network devices. In a bus topology, as in Figure 3.15, all computers or stations attach directly to a bus network through the appropriate hardware interfacing, known as a tap. Full duplex operation between the station and tap allows data to transmit onto the bus and received from the bus. Any station's transmission travels the length of the medium in both directions and can be received by all other stations. At each end of the bus is a terminator, which absorbs any signal, removing it from the bus. Drop lines and taps connect the nodes to the bus cable. A drop line is a connection running between the device and the main cable. A tap is a connector that either splices into the main cable or punctures the sheathing of a cable to create contact with the metallic core. A bus network works best with a limited number of computers.

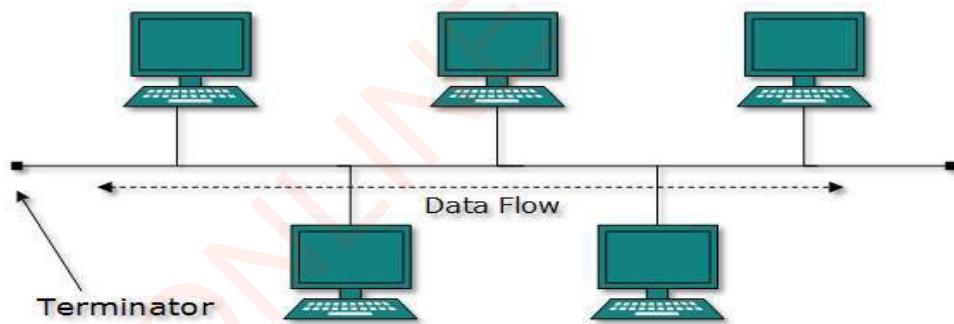


Figure 3.15: Bus topology

#### Advantages of bus topology

- It is an easy installation type of topology.
- Bus topology utilizes few amounts of cables in comparison to other types of topologies.
- Bus topology is suitable for a small network.
- If one computer fails in the network, the other computers are not affected and will continue to work.

- (e) It is also less expensive than a star topology.

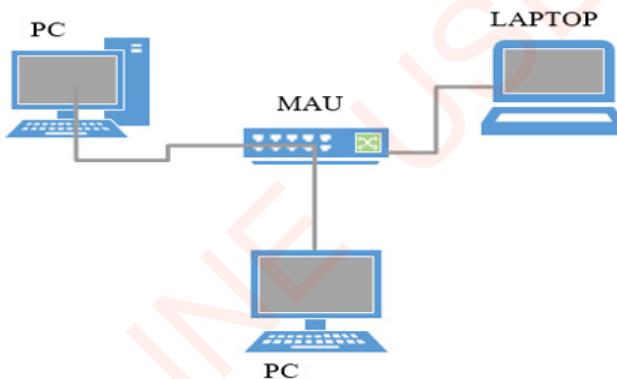
#### *Disadvantages of bus topology*

- The cable length will be limited, limiting the number of stations.
- If the backbone cable fails, the entire network will go down.
- It is very difficult to troubleshoot.
- Maintenance cost is very high in the long run.
- Terminators are required for both ends of the cable.

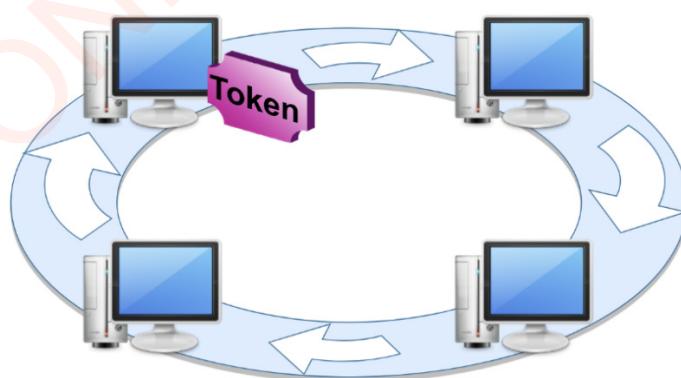
#### *Ring topology*

On this type of topology, as shown in Figure 3.16 (a) and (b), the network consists of a dedicated point-to-point

connection and a set of repeaters in a closed loop. A token is passed along the ring in one direction, from device to device, until it reaches its destination. It may be clockwise or anticlockwise. When a device receives a signal intended for another device, its repeater generates the bits and passes them along. For the bus and tree, data are transmitted in frames. As a frame circulates past all the other stations, the destination station recognizes its address and copies the frame into a local buffer as it goes by. The frame continues to circulate until it returns to the source station, where it is removed. These topologies are used in school campuses and some office buildings.



**Figure 3.16(a): Ring topology, physical representation (PCs are connected to MAU)**



**Figure 3.16(b): Ring topology, electrical representation**

*Advantages of ring topology*

- (a) Relatively easy to install and reconfigure.
- (b) Fault isolation is simplified.
- (c) It performs better than star topology under a heavy workload.
- (d) There is no need for the network server to manage the connection between the computers.
- (e) It is cheaper than star topology because of less wiring.
- (f) One can create a large network by adding the token ring to the network.
- (g) It is an ordered network because all the devices have access to the token ring and the opportunity to transmit.

*Disadvantages of ring topology*

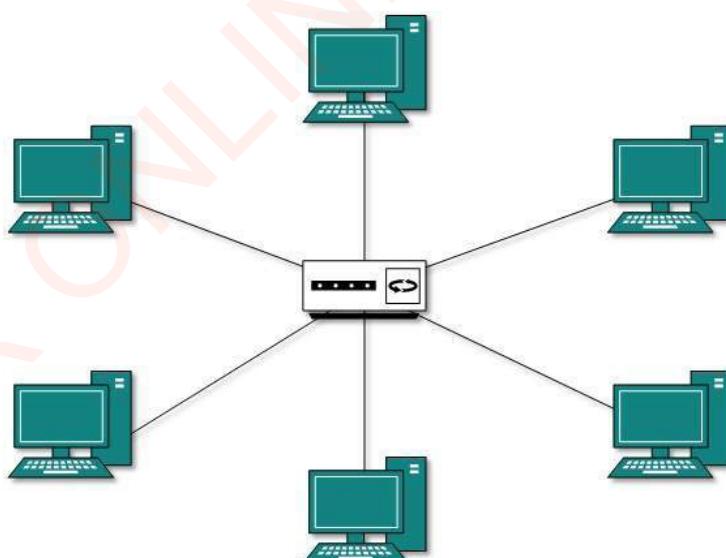
- (a) A break in the ring (such as a disabled station) can disable the entire network.
- (b) It is much slower than an Ethernet network under normal load.

(c) Any moves, changes and adds of the devices can affect the network.

(d) Network connection devices (like Network adapter cards and MAU) are much more expensive than Ethernet cards.

***Star topology***

Each station in a star topology is directly connected to a central node known as the hub/switch. The devices are not physically connected to one another, unlike mesh topology. Direct traffic between devices is not possible in a star topology. When one device wishes to transfer data to another, it sends it to the controller, which then, relays it to the connected device. Each device in a star only requires one connection and one I/O port to connect to any number of others. Local area networks (LANs) and high-speed LANs frequently adopt a star architecture with a central hub. Figure 3.17 shows star topology.



**Figure 3.17: Star topology (the central device is the hub)**

*Advantages of star topology*

- (a) If one link fails in the star topology, only that link is affected. All other links remain active.
- (b) It is easy to identify the fault and fault isolation.
- (c) It is easy to expand the network in the star topology.
- (d) There are no network disruptions when connecting or removing devices from the network.
- (e) It is very easy to manage because of its simplicity of function.

*Disadvantages of star topology*

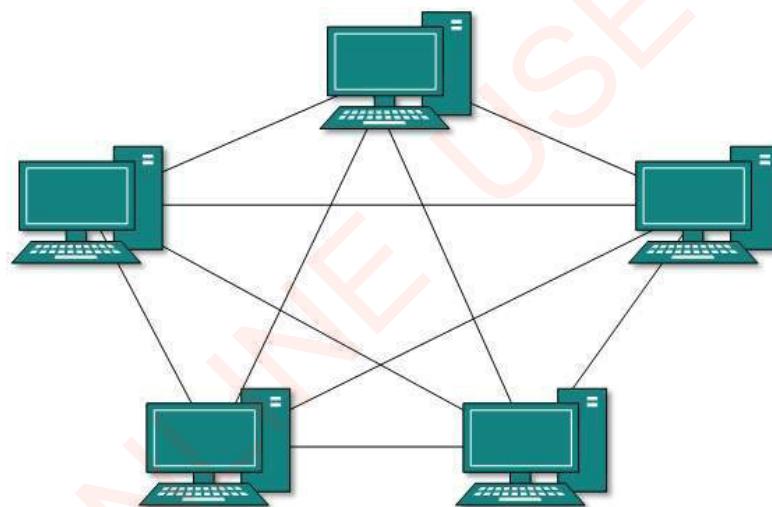
- (a) The entire network will fail if the

central networking device (hub) goes down.

- (b) It requires more cable length compared to the linear bus topology.
- (c) It is much more expensive than bus topology because of the cost of the hubs.

**Mesh topology**

Every device in a mesh topology has its dedicated point-to-point link to every other device. The term “dedicated” refers to the link that exclusively transports data between the two devices it links. Figure 3.18 shows mesh topology.



**Figure 3.18: Mesh topology**

**Types of mesh topology**

**Partially Mesh:** Not all hosts have a point-to-point connection to every other host. Hosts connect to each other in some arbitrary fashion. This topology exists where we need to provide reliability to some hosts out of all.

**Full Mesh:** All hosts have a point-to-point connection to every other host in the network. Thus, for every new host ' $n(n-1)/2$ ', connections are required. Full mesh provides the most reliable network structure among all network topologies.

To calculate the number of physical links in a fully linked mesh network with  $n$  nodes, we must first consider that each node will be connected to at least one other node.

Node 1 must connect to  $n-1$  nodes, node 2 must connect to  $n-1$  nodes, and node  $n$  must connect to  $n-1$  nodes. However, if each physical link can communicate in both directions, the total number of links can be reduced by two. To put it in another way, we require  $n(n-1)/2$  in a full mesh topology.

**Example 3** Calculate number of cables required to connect 4 computers in a full mesh topology.

Solution

$$N_c = n(n-1)/2$$

where  $N_c$  = Number of cables

$n$  = number of nodes

$$n = 4$$

$$N_c = 4(4-1)/2$$

$$= 4 \times 3/2 = 6$$

$$N_c = 6$$

The number of cable required to connect four (4) computers in a full mesh topology is 6 cable.

**Example 4** If you are connecting 13 nodes in a full mesh topology, what will be the required number of cables?

Solution:

$$N_c = n(n-1)/2$$

Where

$N_c$  = Number of cables

$n$  = Number of nodes

$$= 13(13 - 1)/2 = 78$$

Therefore, the total number of cables required for connecting 13 nodes = 78

### Advantages of full mesh

- There is no traffic problem because of the dedicated link in the mesh network.
- Mesh topology is robust. If one link becomes unusable, it does not hinder the entire system.
- Point-to-point links make full identification and fault isolation easy.
- Security or privacy for data travels along the dedicated line.
- The network can be expanded without any disruptions to the users.

### Disadvantages of full mesh

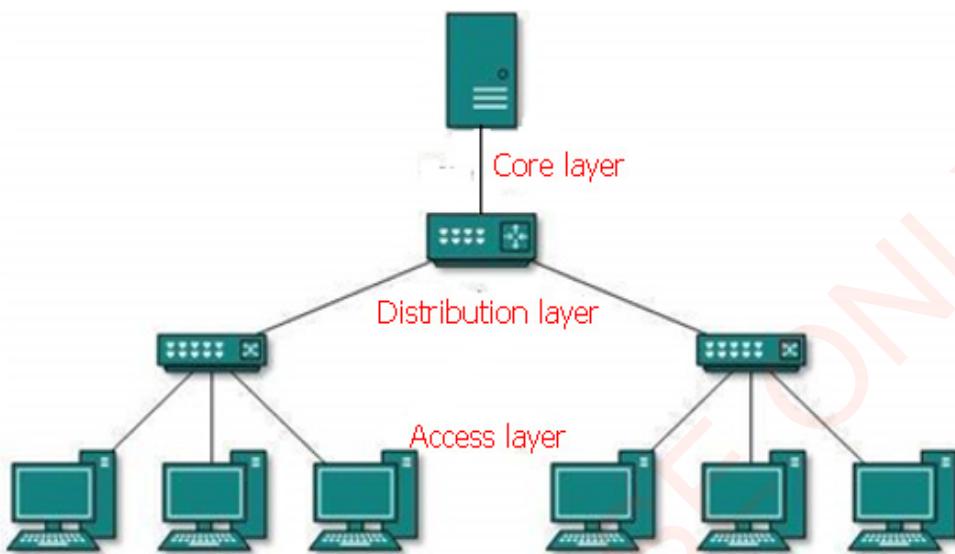
- Installation and reconnection are difficult
- A large amount of cabling and a number of I/O ports is required
- The sheer bulk of wiring can be greater than the available space it can accommodate
- The hardware needed to connect each link can be prohibitively expensive

### Tree topology

This is the most popular type of network topology in use nowadays. It is also known as *hierarchical topology*. This topology imitates extended star topology and inherits bus topology traits.

This network topology divides the network into numerous levels/layers. A network is divided into three layers: the lowest layer is the access-layer, where computers are connected; the middle

layer, which is known as the distribution layer, and it serves as a bridge between the top and lower layers; and the uppermost layer, which is known as the core layer, and it serves as the network's central point, i.e. the root of the tree from which all nodes diverge. Figure 3.19 shows tree topology.



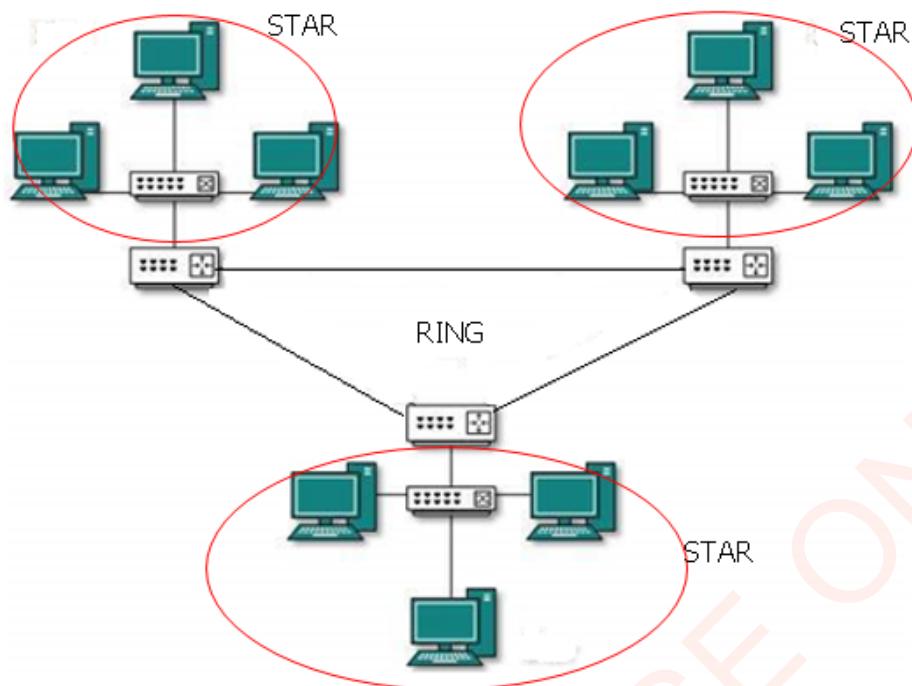
**Figure 3.19:** Tree topology

All nearby hosts are linked together via a point-to-point connection. Similar to the bus topology, if the root fails, the entire network suffers. Despite the fact that it is not the only point of failure, every link serves as a point of failure, and its failure separates the network into inaccessible segments.

### ***Hybrid topology***

A hybrid topology is a network structure designed with more than one topology. Hybrid topology inherits the benefits and drawbacks of all incorporated topologies. Figure 3.20 shows the hybrid topology.

The combining topologies can have characteristics of the star, ring and bus topologies. Most WANs are connected using Dual-Ring topology and the networks that connect to them are largely Star topology networks. The Internet is the most prominent example of a huge hybrid topology.

**Figure 3.20:** Hybrid topology**Activity 3.2: Configure a star network topology**

**Materials:** 5 UTP cables, switch/hub, printer, A4 printing papers, two laptops, and two desktop computers

**Procedure:**

1. Take UTP cables and connect each to a computer and node (hub/switch) using RJ45 Ethernet ports.
2. Switch on the devices and set the IP address so that all nodes are in the same network.
3. Set the printer as a shared one so that every user can print from his/her computer.
4. Load the papers into the printer and test if you can print from each connected computer.

5. Create a folder into one computer and make it shared. Then, try to access the shared folder from each computer.
6. Draw and label the network you have designed and configured.

**Network architecture****Concept of network architecture**

Computer network architecture refers to the design and implementation of a computer network. It is the physical and logical organization and arrangement of various network devices (clients such as PCs, desktops, laptops, mobiles, and so on) to meet the needs of the end user/customer. In this part, you will learn in detail two classifications of well-known

computer network architectures, which are:

- (a) Peer-to-peer
- (b) Client-server network

### Peer-to-peer network architecture

In this network, no server is used for control; every node is itself a client and server. Each node can request and respond for the services and share resources.

In peer-to-peer network architecture, individual devices are directly linked together and have equal obligations and powers in the absence of any central authority. Since no central device is in charge of tasks, this design is sometimes referred to as decentralized architecture. Each computer has particular rights for resource sharing. Nevertheless, if the computer with the resource is unavailable, this may cause problems. This type of architecture is useful in a small environment with few computers. Figure 3.21 shows how peer-to-peer network architecture is connected.



**Figure 3.21:** Peer-to-peer network architecture

### Advantages of peer-to-peer network

- (a) No particular device is a client or a server; the tasks and responsibilities of servers are distributed among all the devices (which also act as clients).
- (b) This network architecture is inexpensive to set up as there is no requirement for a centralized server. It also ensures that, in case of any failure in the network, all unaffected devices continue to operate normally.
- (c) It is simple to set up and maintain because each computer runs independently.

### Disadvantages of peer-to-peer network

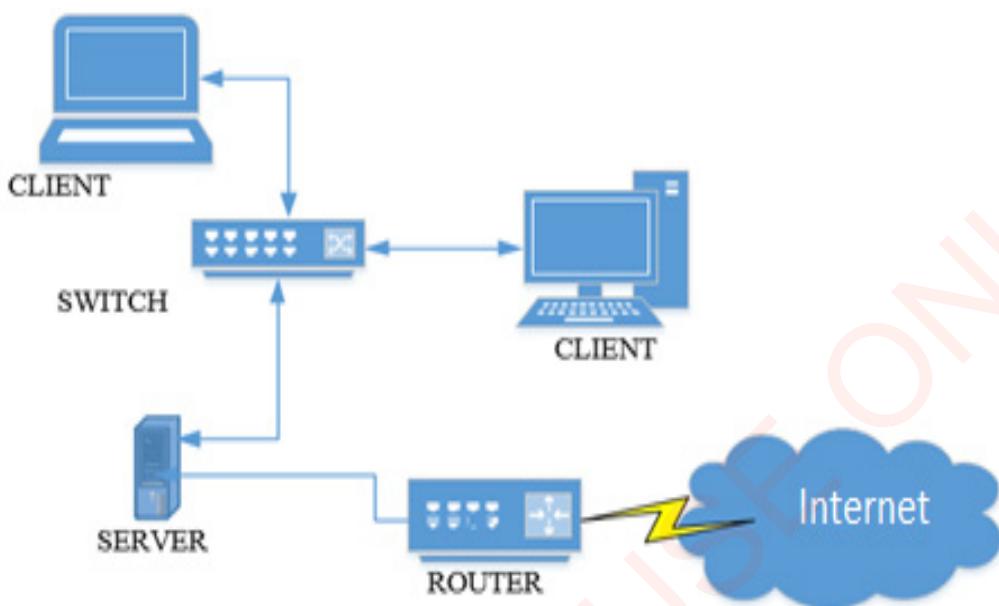
- (a) No centralized system, thus difficult to keep a backup of the data in case of any fault.
- (b) It has a security flaw because computers are self-managed.
- (c) With a growth in the number of nodes on this network, performance, security, and access may all become more challenging.

### Client-server network architecture

It is a computer networking model that comprises of a powerful central computer that controls and provides services to all terminals/client PCs connected to it. This central computer is called a server. Client computers connect to the server only when they need to use shared resources or data. All exchanged data is saved entirely on the server, not on any other machine.

The server computer does all types of processing and controlling in

this networking arrangement. A server is a specialized computer that handles all key functions, such as security and network management, as well as controlling network resources and providing services to other computers on the network. A server provides the desired services to a client computer, as shown in Figure 3.22.



**Figure 3.22:** The client-server architecture

### Classification of client/server computing network architecture

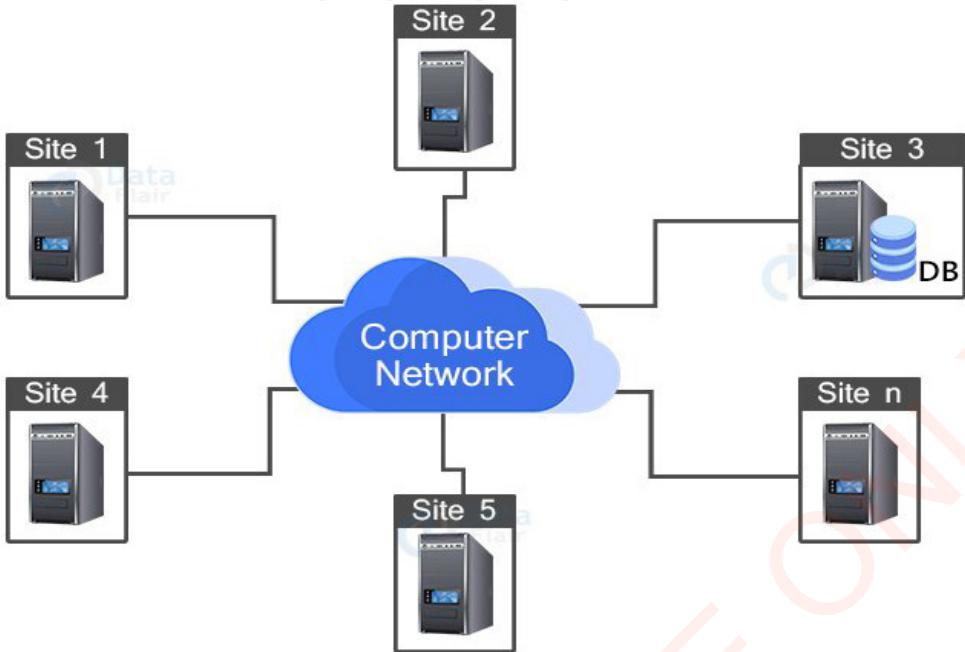
Depending on the number of servers and the storage media, client/server network architecture can be further divided into two types.

- (a) Centralized Client-Server Network architecture.
- (b) Distributed Client-Server Network architecture.

**Centralised Client/Server computing architecture:** Centralized network architecture is a client/server architecture system in which one or more client

nodes are directly connected to a server computer. This is the most popular system used in many organizations, in which a client submits a request to a server and receives responses.

In other words, centralized architecture is one of client/server architecture having one powerful computer utilized to service one or more low-powered computers in centralized computing architecture. The nodes under the centralized architecture are not linked; they are only connected to the server. Figure 3.23 shows the centralized computing architecture.



**Figure 3.23:** Centralised client-server computing architecture

#### *Components of centralised client-server architecture*

The centralised client-server architecture includes the following components:

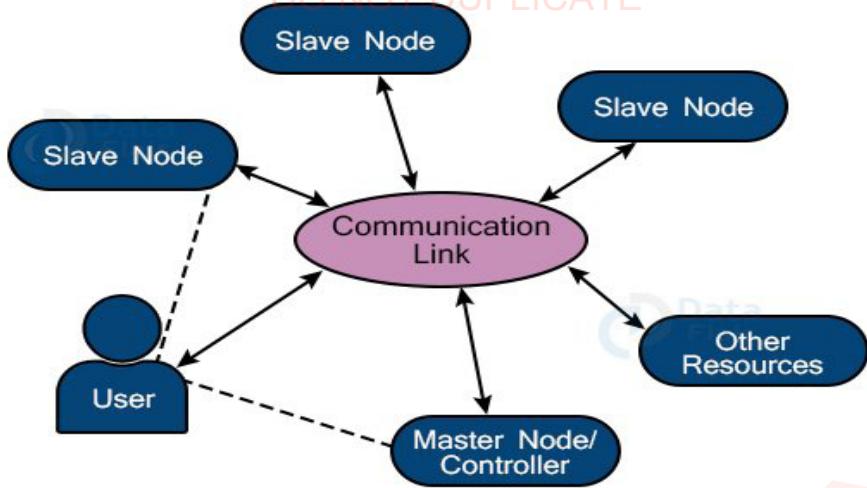
- (a) Server computer/main frame computer: The powerful computer which handles all processing.
- (b) Terminals: These are computer connected to a central computer and function as input/output devices. Linking of at least two mainframe computers together via networks, terminals communicate solely with the mainframe and never with one another.

*Distributed Client/Server architecture:* There is a mainframe server that manages other servers in the network that are

connected to it. It supports various functions, including file, hardware, and network sharing. Client-server computers process their own data and transmit the most recent modifications to the mainframe. The mainframe server computer handles some of the processes. Figure 3.24 illustrates distributed computing architecture.

The following features are found in distributed client-server architecture:

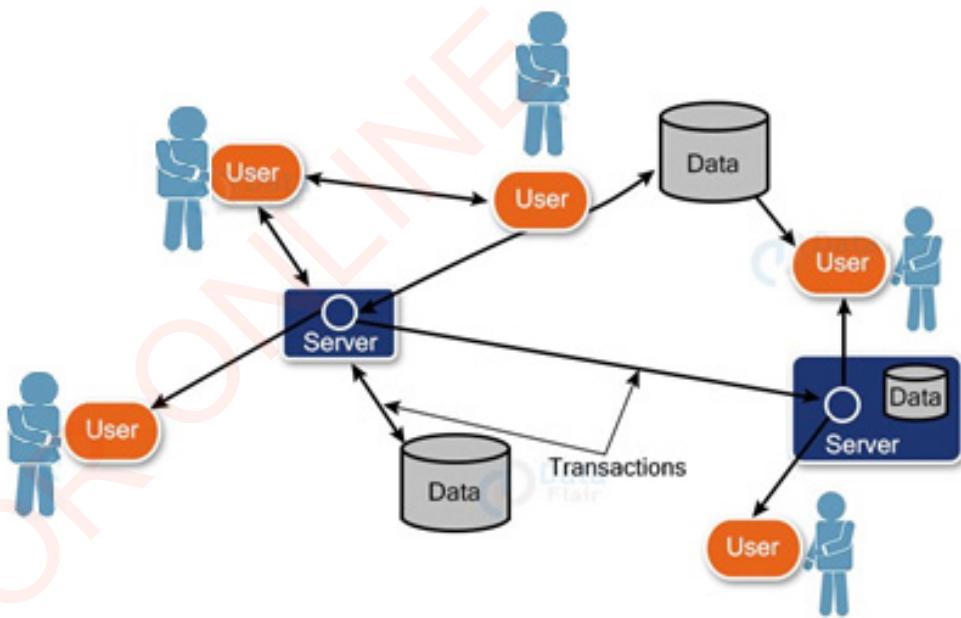
- (i) Different computers are effective at performing independently
- (ii) Completion of tasks on multiple computers locally
- (iii) Networks enable computers to exchange data and services but do not offer processing help.



**Figure 3.24:** Distributed client-server computing architecture

### **Collaborative computing architecture**

Collaborative computing architecture is a hybrid of centralized and distributive computing architecture. Under the collaborative paradigm, individual network members can process their users' basic needs. Figure 3.25 shows collaborative computing architecture.



**Figure 3.25:** Collaborative computing architecture

**Table 3.4:** Difference between Client-Server and Peer-to-Peer architecture

| Basis of comparisons    | Client-server architecture                                                                                                                                                               | Peer-to-peer architecture                                                                                      |
|-------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------|
| Focus area              | Information sharing                                                                                                                                                                      | Connectivity                                                                                                   |
| Data                    | Data are stored in the central server in the case of a centralized server, whereas, in the case of the distributed server, data are stored on various servers located on different sites | Each peer has its own data                                                                                     |
| Implementation          | Expensive compared to a peer-to-peer network                                                                                                                                             | Less expensive to implement                                                                                    |
| Differentiation         | Client and server are differentiated; specific servers and clients are presents                                                                                                          | Client and server are not differentiated                                                                       |
| Stability               | More stable and scalable                                                                                                                                                                 | Less stable if the number of peers increases                                                                   |
| Services                | The client request service from the server and the server responds with a service                                                                                                        | Each node can do both requests and responses for the services                                                  |
| Security                | More secure than a peer-to-peer network because the server can authenticate a client's access to any part of a network.                                                                  | It is less secure compared to a Client-server network                                                          |
| Performance             | Does not experience performance problems with the increase of clients since most of the heavy tasks are carried by sever                                                                 | The increase in clients will affect the performance of the network                                             |
| Reliability             | It is less reliable compared to Peer-to-Peer                                                                                                                                             | It is more reliable                                                                                            |
| Access time for service | Multiple clients request services from the server and therefore, access time for service is high                                                                                         | Service providing nodes are distributed; for that reason, a service-requesting note does not need to wait long |

### ***Advantages of Client-Server architecture***

- (a) This type of architecture is much easier to scale up since it is much more convenient to add more server computers than configure the network on each computer (as in peer-to-peer architecture).

- (b) It has a much faster network speed compared to peer-to-peer network architecture
- (c) This network architecture is more secure than peer-to-peer because, in this type of architecture, a single server manages the shared resources in a Client/Server network
- (d) Backing up data is easy because of the centralized system.
- (e) The server provides a customized Network Operating System (NOS) to offer resources to multiple users who require them.

### ***Disadvantages of Client-Server architecture***

- (a) It is more prone to downtime because, if the server fails, none of the client machines can get their requests served.
- (b) Require a dedicated network administrator to handle all of the resources.
- (c) It is far more expensive than P2P. This is due to the requirement for a server with more RAM as well as the necessity for several networking devices such as hubs, routers, switches, and so on.

### **Exercise 3.2**

1. Explain different types of computer networks.
2. Describes six networking devices used in data communication.
3. What is network topology? Discuss the classification of network topologies. Contrast between Tree topology and Mesh topology.

4. Assume six devices are arranged in a Mesh full topology, then
- (a) Calculate number cables that will be required to make the topology.
  - (b) How many ports will be required for each device?
5. Describe in details merits and demerits of distributed network architecture over centralized architecture.



### **Activity 3.3: Peer-to-peer file and folder sharing**

Follow the following steps to share your file or folder on your network

Step 1: Navigate to the desktop

Step 2: Create your folder

Step 3: Navigate to the folder and open the properties

Step 4: Choose who you want to share with

Step 5: Share the folder

Step 6: Permissions: Make sure you give full control to the people that have access to the shared folder. Click Apply, then click OK. Once you press OK, you will be back at the advanced sharing page. Press Apply and OK on that page too.

Step 7: Open Control Panel

Step 8: Network and sharing

Step 9: Navigate to the advanced sharing settings

Step 10: Choose Home and Work / Public

Step 11: Select all options

Step 12: Navigate to networks: Go onto another computer and open the file

explorer. Go into the Network section found on the left hand side at the bottom

Step 13: Find the device: Find the original device that the file was shared from

Step 14: Find the folder that was shared

Step 15: Open the folder to see the file contained in it

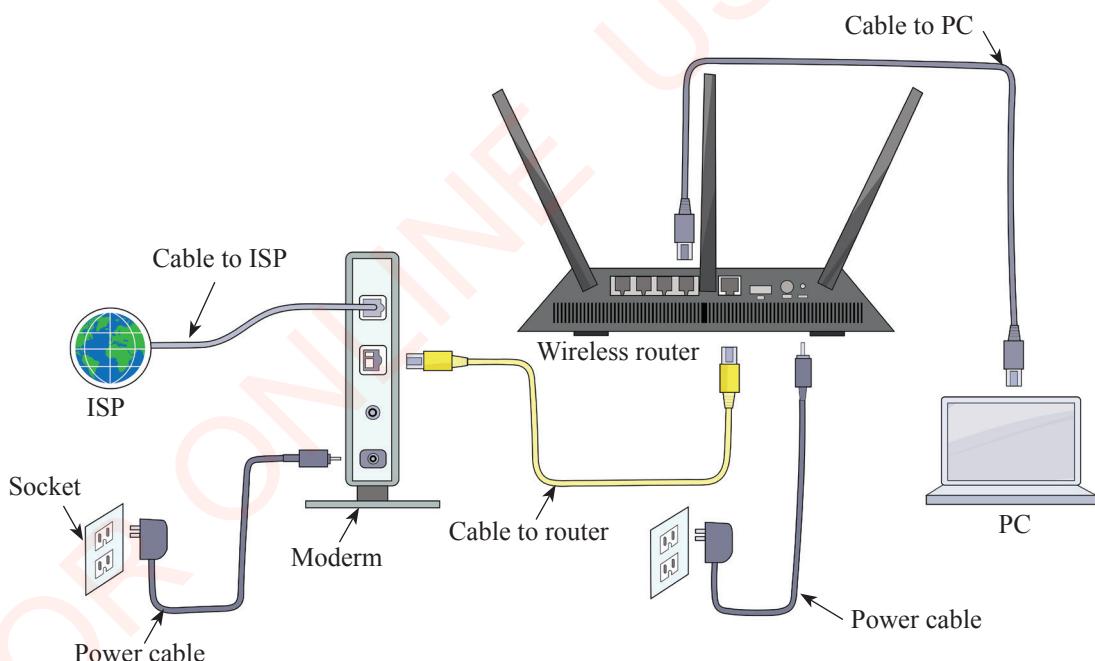
Step 16: Done

on the computer is digital, whereas a telephone line or cable wire can transmit only analog data.

The primary function of the modem is to convert the digital signal into analog and vice versa. A modem is a combination of two devices modulator and a demodulator. The modulator converts digital data into analog data when the computer is sending the data. The demodulator converts analog signals into digital data when the computer receives it.

#### *Types of modem*

The modem can be categorised in several ways depending on how it can transmit data, the type of connection to the transmission line, transmission mode, etc. Figure 3.26 shows the modem and router connection.



**Figure 3.26:** Modem and router connection

**Repeater:** A repeater is a physical layer device. Its main function is to regenerate the signal over the same network before the signal becomes too weak or corrupted to extend the length to which the signal can be transmitted over the same network.

An important point to be noted about repeaters is that they do not amplify the signal. When the signal becomes weak, they copy it bit by bit and regenerate it at its original strength. It is a 2-port device. Figure 3.27 shows a multipoint repeater.



**Figure 3.27:** Dual multiport repeater

**Hub:** This is a multiport repeater. It connects multiple wires coming from different branches; for example, the connector in a star topology connects different stations. Hubs cannot filter data, so data packets are sent to all connected devices. In other words, the collision domain of all hosts connected through a hub remains one. Also, they do not have the intelligence to find the best path for data packets, leading to inefficiencies and waste. Figure 3.28 shows Hub (D-Link)



**Figure 3.28:** Hub (D-Link)

#### Types of Hubs

**Active Hub:** This type of hub has its own power supply and can clean, boost, and relay the signal along with the network. It serves both as a repeater and as a wiring centre. It is used to extend the maximum distance between nodes.

**Passive Hub:** This is the type of hub that collects wiring from nodes and power

supply from the active hub. These hubs relay signals onto the network without cleaning and boosting them and can not be used to extend the distance between nodes.

**Intelligent Hub:** This is the type of hub that works like active hubs and includes remote management capabilities. It also provides flexible data rates to network devices. Moreover, they enable an administrator to monitor the traffic passing through the hub and configure each port.

**Bridge:** This operates at the data link layer. A bridge is a repeater with add on the functionality of filtering content by reading the MAC addresses of the source and destination. It is also used for interconnecting two LANs working on the same protocol. It has a single input and single output port, thus making it a 2 port device. Figure 3.29 shows the network bridge.



**Figure 3.29:** Example of network bridge

#### Types of Bridges

**Transparent Bridges:** These are bridges in which the stations are completely unaware of the bridge's existence, i.e. whether or not a bridge is added or deleted from the network, thus, reconfiguration of the stations is unnecessary. These bridges use two processes, i.e. bridge forwarding and bridge learning.

**Source Routing Bridges:** In these bridges, routing operation is performed by the source station and the frame specifies which route to follow. The host can discover the frame by sending a special frame called the discovery frame, which spreads through the entire network using all possible paths to the destination.

**Switch:** This network device uses a packet switching technique to receive, store and forward data packets on the network. The switch maintains a list of network addresses of all the devices connected to it. It is sometimes known as an intelligent hub.

Upon receiving a packet, it checks the destination MAC address and transmits the packets to the correct port. Before forwarding, the packets are checked for collision and other network errors. The data is transmitted in full duplex mode. It forwards data based on the MAC address.

Data transmission speed in switches can be double that of other network devices like hubs used for networking. This is because a switch shares its maximum speed with all the connected devices. This helps to maintain the network speed even during high traffic. In fact, higher data speeds are achieved on networks through the use of multiple switches. Figure 3.30 shows the Switch with 24 ports.



**Figure 3.30:** Switch with 24 ports

**Routers:** This is a device like a switch that routes data packets based on their IP addresses. The router is mainly

a Network Layer device. Routers normally connect LANs and WANs together and have a dynamically updating routing table based on which they make decisions on routing the data packets. The router divides the broadcast domains of hosts connected through it. A router also strengthens the signals before transmitting them; that is why it is also called a repeater. Figure 3.31 shows the Wireless router.



**Figure 3.31:** Wireless router

**Routing Table:** A router reads its routing table to decide the best available route the packet can take to reach its destination quickly and accurately. The routing table may be of these two types:

**Static** – In a static routing table, the routes are fed manually. So, it is suitable only for very small networks with a maximum of two to three routers.

**Dynamic** – In a dynamic routing table, the router communicates with other routers through protocols to determine free routes. This is suited for larger networks where manual feeding may not be feasible due to a large number of routers.

**Gateway:** A gateway, as the name suggests, is a passage to connect two networks together that may work upon different networking models. They basically work as messenger agents that take data from one system, interpret

it and transfer it to another system. Gateways are also called protocol converters and can operate at any network layer. Gateways are generally more complex than switches or routers. Figure 3.32 shows a Gateway.



**Figure 3.32:** Gateway

**Brouter:** This is also known as the bridging router, which is a device that combines features of both bridge and router. It can work either at the data link layer or a network layer. Working as a router, it can route packets across networks, and when working as a bridge, it can filter local area network traffic. Table 3.5 shows the difference between Router and Brouter.

**Table 3. 5:** Difference between Router and Brouter

| S.No | Router                                                                                | Brouter                                                                                                                 |
|------|---------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------|
| 1.   | A router is a networking device that forwards data packets between computer networks. | A Brouter is a networking device used both as a bridge and a router; it combines a network bridge and a router.         |
| 2.   | It operates at a Network layer.                                                       | It operates either at a Data link layer or a Network layer.                                                             |
| 3.   | The router stores the routing table.                                                  | Brouter stores the routing table when configured as a router and stores the MAC address as a bridge.                    |
| 4.   | It takes forwarding decisions based on IP address.                                    | Forwarding decisions are based on IP address when configured as a router or MAC address when configured as a bridge.    |
| 5.   | The router transmits data in the form of packets.                                     | Brouter transmits data in the form of packets when it is configured as a router and frames when configured as a bridge. |
| 6.   | The router works on more than one broadcast domain.                                   | Brouter works on more than one broadcast domain when configured as a router and a single broadcast domain as a bridge.  |
| 7.   | The router is a full duplex.                                                          | Brouter is a full duplex when configured as a router and a half-duplex as a bridge.                                     |

## *Network interface card (NIC) NOT DUE Wi-Fi card*

This is the network adapter that is used to connect the computer to the network. The earliest network cards were external to the system and needed to be installed manually to establish a network. In modern computer systems, it is an internal hardware component.

NIC has a unique ID (MAC address/physical address) written on the chip, and it has a connector to connect the cable. The cable acts as an interface between the computer and the router or modem. NIC card is a layer 2 device that works on the network in the physical and data link layers.

### *Types of network card*

There are two types of network cards namely: Ethernet Card for Ethernet connection and wireless network card (Wi-fi) for wireless network connection.

*Ethernet card:* To establish a network connection using wire, the device should have an Ethernet card to support the connection. The NIC has an RJ45 socket where the network cable is physically plugged in.

*Ethernet card speeds* may vary depending on the protocols it supports. Old Ethernet cards had a maximum speed of 10 Mbps. However, modern cards support fast Ethernets up to a speed of 100 Mbps. Some cards even have a capacity of 1 Gbps. This is generally known as bandwidth speed. The Ethernet Card is shown in Figure 3.33.



**Figure 3.33: The Ethernet card**

*Wi-Fi* is the acronym for wireless fidelity. Wi-Fi technology is used to achieve wireless connection to any network. A Wi-Fi card is a card used to connect any device to the local network wirelessly. The physical area of the network which provides internet access through Wi-Fi is called a Wi-Fi hotspot. Hotspots can be set up at the home, office or public space. Hotspots themselves are connected to the network through wires. The wireless Network card is shown in Figure 3.34.



**Figure 3.34: Wireless Network card**

A Wi-Fi card is used to add capabilities like teleconferencing, downloading digital camera images, video chat, etc., to old devices. Modern devices come with their in-built wireless network adapter.

## **Networking models**

There are two types of network models: the OSI Model and the TCP/IP model. This section will describe the concept behind these models with respect to the data transmission process in the network.

### **Open System Interconnection (OSI) model**

OSI, stands for Open System Interconnection, is a reference model

that describes how information from a software application in one computer moves through a physical medium to the software application in another computer.

OSI model describes seven layers that computer systems use to communicate over a network. Each layer performs a particular task. It was the first standard model for network communications, adopted by all major computer and telecommunication companies in the early 1980s.

The International Organization for Standardization (ISO) developed the OSI model in 1984, and it is now considered an architectural model for

inter-computer communications. OSI model divides the whole task into seven smaller and more manageable tasks. Each layer is assigned a particular task and each layer is self-contained so that the task assigned to each layer can be performed independently.

The modern internet is established based on the TCP/IP model, a simplified version of the OSI model. However, the OSI 7-layer model is still widely used, as it helps visualize and communicate how networks operate and helps to isolate and troubleshoot networking problems. Table 3.6 shows OSI model layers, their functions and examples.

**Table 3.6: OSI Model layers and their function**

| Layer            | Function                                                                     | Example                   |
|------------------|------------------------------------------------------------------------------|---------------------------|
| Application (7)  | Services that are used with end user applications                            | SMTP                      |
| Presentation (6) | Formats the data so that it can be viewed by the user<br>Encrypt and decrypt | JPG, GIF, HTTPS, SSL, TLS |
| Session (5)      | Establishes/ends connections between two hosts                               | NetBIOS, PPTP             |
| Transport (4)    | Responsible for the transport protocol and error handling                    | TCP, UDP                  |
| Network (3)      | Reads the IP address from the data packet                                    | Routers, Layer 3 Switches |
| Data Link (2)    | Reads the MAC address from the data packet                                   | Switches                  |
| Physical (1)     | Send data on to the physical wire                                            | Hubs, NICs, Cable         |

### **Characteristics of OSI layers**

The OSI model is divided into two layers; upper layers and lower layers.

The upper layer of the OSI model mainly deals with application-related issues, which are implemented only in the application software. The application layer is closest to the end user such that the end user uses the application layer to interact with the software applications.

The lower layer of the OSI model deals with data transport issues. The data link and physical layers are implemented in hardware and software. The physical layer is the lowest layer of the OSI model and is closest to the physical medium. The physical layer is mainly responsible for placing the data on the physical medium.

### **Functions of OSI layers**

**Layer 7-Application:** The application layer is used by end-user software such as web browsers and email clients. It provides protocols that allow application software to send and receive data and present meaningful information to users. A few examples of application layer protocols are the Hypertext Transfer Protocol (HTTP), File Transfer Protocol (FTP), Post Office Protocol (POP), and Simple Mail Transfer Protocol (SMTP).

**Layer 6-Presentation:** The presentation layer prepares data for the application layer. It defines how two devices should encode, encrypt, and compress data to receive it correctly on the other end. The presentation layer takes any data transmitted by the application layer and prepares it for transmission over the session layer.

**Layer 5-Session:** The session layer creates communication channels, called sessions, between devices. It is responsible for opening sessions, ensuring they remain open and functional while data is being transferred, and closing them when communication ends. The session layer can also set checkpoints during a data transfer if the session is interrupted, devices can resume data transfer from the last checkpoint.

**Layer 4-Transport:** The transport layer takes data transferred in the session layer and breaks it into “segments” on the transmitting end. It is responsible for reassembling the segments on the receiving end, and turning them back into data that the session layer can use. The transport layer carries out flow control, sending data at a rate that matches the connection speed of the receiving device, and error control, checking if data was received incorrectly; and, if not, requesting it again. There are two main protocols under this layer.

#### **(a) Transmission Control Protocol**

Transmission Control Protocol is responsible for the systems to communicate over the internet. It establishes and maintains a connection between hosts when sending the data. In order to send the data, the message to be sent will be divided into smaller units regarded as segments. During transmission, each segment is transmitted through multiple routers, once they reach the destination, TCP reorders the packets in the correct order.

#### **(b) User Datagram Protocol (UDP)**

User Datagram Protocol is a transport layer protocol. It is an unreliable transport protocol since, in this case, the receiver does not send any acknowledgement when the packet is received, and the sender does not wait for any acknowledgement. Therefore, this makes a protocol unreliable.

**Layer 3-Network:** The network layer has two main functions. One is breaking up segments into network packets, and reassembling the packets on the receiving end. The other is routing

packets by discovering the best path across a physical network. The network layer uses network addresses (typically Internet Protocol addresses) to route packets to a destination node.

*Layer 2-Data Link:* The data link layer establishes and terminates a connection between two physically-connected nodes on a network. It breaks up packets into frames and sends them from source to destination. This layer comprises two parts: Logical Link Control (LLC), which identifies network protocols, performs error checking and synchronizes frames, and Media Access Control (MAC), which uses MAC addresses to connect devices and defines permissions to transmit and receive data.

*Layer 1-Physical:* The physical layer is responsible for the physical cable or wireless connection between network nodes. It defines the connector, the electrical cable or wireless technology connecting the devices, and is responsible for transmitting the raw data, which is simply a series of 0s and 1s, while taking care of bit rate control.

#### **Advantages of OSI Model**

The following are the advantages of the OSI model :

- The OSI Model helps the users and operators of computer networks to;
  - determine the required hardware and software to build their network.
  - understand and communicate the process followed by components communicating across a network.
  - perform troubleshooting by

identifying which network layer is causing an issue and focusing efforts on that layer.

- The OSI Model helps the network device manufacturers and networking software vendors to;
  - create devices and software that can communicate with products from any other vendor, allowing open interoperability.
  - define which parts of the network their products should work with.
  - communicate to users at which network layers their product operates.

#### **TCP/IP Model**

The TCP/IP model is sometimes called *TCP/IP protocol suite*. It is a suite of communication protocols used to interconnect network devices on the internet. This model was designed and developed by the US Department of Defense (DoD) in the 1960s and is based on standard protocols. It stands for Transmission Control Protocol/Internet Protocol. It contains four layers, unlike the seven layers in the OSI model. The following are the TCP/IP protocol suite layers:

- Application layer
- Transport layer
- Internet layer
- Network Access layer

The diagrammatic comparison of the TCP/IP and OSI model is as in Figure 3.35.

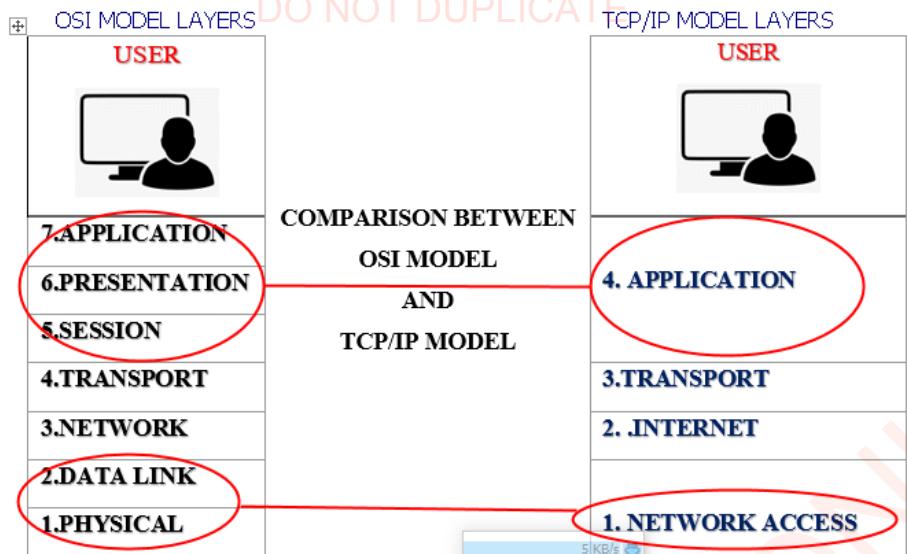


Figure 3.35: TCP/IP vs OSI Model layers

### Layer 1. Network Access

This layer combines the Data Link Layer and the Physical Layer of the OSI model. The protocols in this layer support hardware addressing to allow for the physical transmission of data.

### Layer 2. Internet

An Internet layer is the second layer that is also known as the network layer with the main responsibility of sending the packets from any network to the destination irrespective of the route they take. The following are the protocols used in this layer:

**IP Protocol:** IP protocol is the most significant part of the entire TCP/IP suite with the following responsibilities:

**IP Addressing:** IP protocol implements logical host addresses known as IP addresses, which are used by the internet and higher layers to identify the device and provide internetwork routing.

**Host-to-host communication:** It uses this function to determine the path through which the data is transmitted.

**Data encapsulation and formatting:** It uses this function to encapsulate the data into a message format known as an IP datagram to ensure data is sent and received securely.

**Fragmentation and reassembly:** This function splits the IP datagram into smaller units if the size of the datagram is greater than the Maximum Transmission Unit (MTU), which is a limit imposed by the datalink layer protocol. On the receiving side, the function reassembles fragmented units to form the original datagram.

**Routing:** When an IP datagram is sent over the same local network, such as LAN, MAN, or WAN, it is known as direct delivery. The IP datagram is sent indirectly when the source and destination are on a distant network. This can be accomplished by routing

the IP datagram through various devices such as routers.

### *ARP Protocol*

Address Resolution Protocol (ARP) is a network layer protocol with the function of finding physical addresses from IP addresses. ARP protocol contains ARP requests and ARP replies.

*ARP request:* ARP request is used to find the device's physical address.

*ARP reply:* Nodes within a network will process the request from ARP request, and the receiving device will understand the request and reply to the device's physical address in the form of an ARP reply.

### *Layer 3. Transport*

Its main function is to handle end-to-end communication and error-free delivery of data. Transport layers have two main protocols:

*Transmission Control Protocol (TCP):* TCP ensures reliable and error-free communication between the end system. It performs sequencing and segmentation of data. It is a very effective protocol but has a lot of overhead due to such features. Increased overhead leads to increased costs.

*User Datagram Protocol (UDP):* Unlike TCP, UDP is not considered reliable and error-free.

### *Layer 4. Application*

The application layer functions are similar to the top three OSI model layers: Application, Presentation, and Session. The application layer is used to manage node-to-node communication and user-interface specifications. Protocols found in application layers include HTTP, FTP,

DHCP, TFTP, SMTP, SNP and DHCP.

### *TCP and UDP mechanism*

#### *Transmission Control Protocol (TCP)*

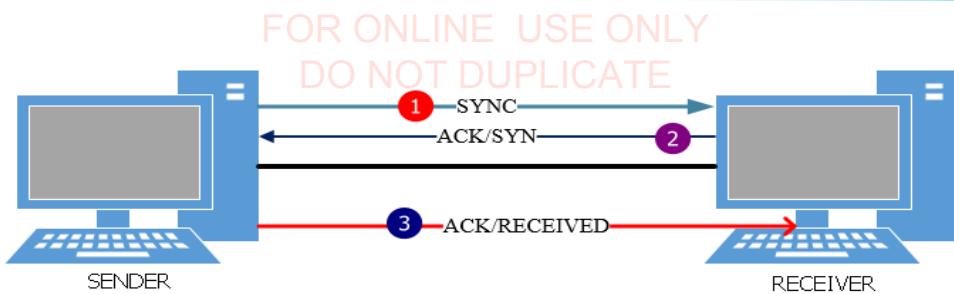
TCP is a communication protocol that enables application programs and computing devices to exchange messages over a network. It is designed to send packets across the Internet and ensure the successful delivery of data and messages over networks.

#### *TCP Features*

The following are the features of the TCP:

(i) *Data delivery:* TCP ensures that the data is received correctly, no data is missing, and it reaches the destination in the order it was sent. If TCP is not used, incorrect data can be received or be out of order. For example, some data or images can be missing if we try to view the web page or download a file without TCP.

(ii) *Protocol:* TCP is a connection-oriented protocol which means it establishes a connection before communicating. The process of establishing a connection is done through a three-way handshake process. First, the sender sends the SYN message to the receiver, and the receiver sends back the SYN-ACK message to confirm that the message has been received. After receiving the SYN-ACK message, the sender sends the acknowledgement message to the receiver. In this way, the connection is established between the computers. Once the connection is established, the data will be delivered. With TCP, there is a guarantee that, data will be delivered successfully, if the data is not received, the TCP will resend the data. Figure 3.36 shows the TCP three-way handshake process.



**Figure 3.36:** TCP three-way handshake mechanism

### User Datagram Protocol (UDP)

Similarly to TCP, UDP is also used for sending and receiving messages. Unlike TCP, UDP is a connectionless protocol, meaning it does not need to establish a connection before sending the message. This means, UDP does not guarantee the delivery of the data packet to the receiver side. UDP is also known as the “fire-and-forget” protocol, because it sends the data and does not care whether the data is received or not. Differences between the TCP and UDP are shown in Table 3.7.

**Table 3.7:** The difference between TCP and UDP

|                    | TCP                                                                                                                                                       | UDP                                                                                                             |
|--------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------|
| Type of connection | It is a connection-oriented protocol, meaning the connection needs to be established before the data is transmitted over the network.                     | It is a connectionless protocol that sends the data without checking whether the system is ready to receive it. |
| Reliable           | It is a reliable protocol as it assures the delivery of data packets.                                                                                     | It is an unreliable protocol as it does not guarantee the delivery of packets.                                  |
| Speed              | It is slower than UDP as it performs error checking and flow control and assures the delivery of data packets                                             | It is faster than TCP as it does not guarantee the delivery of data packets                                     |
| Header size        | The size of TCP is 20 bytes.                                                                                                                              | The size of the UDP is 8 bytes                                                                                  |
| Acknowledgement    | It uses a three-way handshake concept. In this concept, if the sender receives the ACK, the sender will send the data. TCP also can resend the lost data. | It does not wait for any acknowledgement rather, it just sends the data.                                        |

|                        |                                                                                                                                     |                                                                                                                                                 |
|------------------------|-------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------|
| Flow control mechanism | It follows the flow control mechanism in which too many packets cannot be sent to the receiver simultaneously                       | This protocol follows no such mechanism                                                                                                         |
| Error checking         | It performs error checking by using a checksum. When the data is corrected, the data is retransmitted to the receiver               | It does not perform any error checking and does not resend the lost data packets                                                                |
| Applications           | It is mainly used where a secure and reliable communication process is required (like military services, web browsing, and e-mail). | It is used where fast communication is required and does not care about reliability, like VoIP, game streaming, video and music streaming, etc. |

### ***Data encapsulation and data de-capsulation process***

***Data encapsulation:*** This is the process of adding additional information when data is processed along layers of OSI or TCP/IP model. The data application software is processed across application presentation, and session layers of the OSI model without any additional information added across these three upper layers.

#### ***Data Encapsulation process***

In the transport layer, the data is broken up into pieces. The transport layer adds the header in each broken data, which contains information such as, source port, destination port, sequence number, etc. So, everything is combined into a new form.

The encapsulated data in the Transport layer is referred as Segments or Datagrams. When TCP is used in

transmission, it is referred to as Segments, while when UDP is used for transmission, it is referred as Datagrams.



The data will travel and reach the Network layer. Additional information, which includes source IP and destination IP, from the layer 3 header is added. A combination of all data is referred as Packets.



The Network layer sends the Packets to the Data Link layer. On reaching the Data Link layer, additional information referred to as trailer, which contains MAC address, destination MAC address etc., is added. Furthermore, the trailer added is also used for error checking. In the Data Link layer, the encapsulated data is referred as Frames.



## FOR ONLINE USE ONLY

The Frame in the Data Link layer is delivered to the physical layer and is referred as Bits.

*Protocol Data Unit (PDU):* The encapsulated data tends to attain different names referred as Protocol Data Unit when travelling into different layers, as shown in Table 3.8.

**Table 3.8:** Protocol Data Unit

| LAYER        | PDU                |
|--------------|--------------------|
| Application  | Data               |
| Presentation | Data               |
| Session      | Data               |
| Transport    | Segments/Datagrams |
| Network      | Packets            |
| Data Link    | Frames             |
| Physical     | Bits               |

### De-encapsulation

De-encapsulation is considered as the opposite process of encapsulation. All information added during encapsulation is removed when travelling on the receiver's side from the physical layer to the application layer.

### Data De-encapsulation process

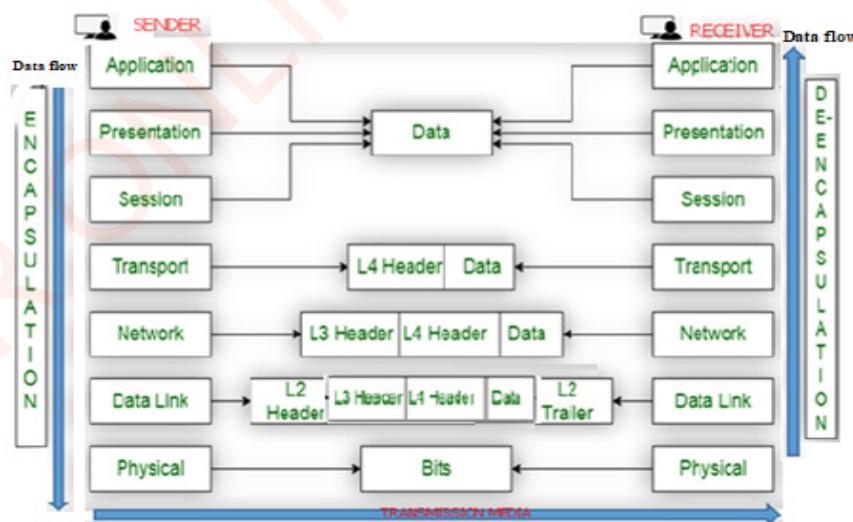
The Physical layer gets the bits, de-encapsulates them into frames, and sends them to the upper layer.

The Data Link layer receives the frames and checks whether the MAC address matches or not. If everything is matched and no error is found, the layer 2 header and layer 2 trailers are removed. It de-encapsulates the data and packet sent to the upper layer.

The Network layer receives the packet from the Data Link layer and checks for IP addresses. If it matches, the header in layer 3 is removed. Now the de-encapsulated data packet is delivered to the Transport layer.

Then, the Transport layer gets the data segments/datagrams from the Network layer and removes the layer 4 header.

After travelling through Session, Presentation, and Application layers, the de-encapsulated data is sent to the recipient. That is how the de-encapsulation takes place: Figure 3.37 data encapsulation and de-encapsulation.



**Figure 3.37:** Data encapsulation and de-encapsulation

### The difference between the TCP/IP model and OSI model

The key difference between the TCP/IP model and the OSI model is that TCP/IP is simpler and combines several OSI layers into one layer. Other differences are:

- OSI layers 5, 6, and 7 are combined into one Application Layer in TCP/IP.
- OSI layers 1, and 2 are combined into one Network Access layer in TCP/IP.
- TCP/IP does not take responsibility for sequencing and acknowledgement functions but leaves these responsibilities to the underlying transport layer.
- TCP/IP is a functional model designed to solve specific communication problems which are based on specific standard protocols. At the same time, OSI is a generic protocol-independent model intended to describe all forms of network communication.
- In TCP/IP, most applications use all the layers, while in OSI, simple applications do not use all seven layers. Only layers 1, 2, and 3 are mandatory to enable any data communication.

### Internet Protocol (IP) addressing

#### Concepts of IP addressing

The Internet Protocol is the cornerstone of the TCP/IP architecture. All computers on the Internet understand IP. The main tasks of IP are addressing the computers and the fragmentation of packets.

**IP Address:** This is a logical numeric address that is assigned to every single computer, printer, switch, router, or any other device that is part of a TCP/IP-based network.

It is also, an identifier since this address is used to identify the device on a network. It defines the technical format of the packets. Mainly, both network models, i.e., IP and TCP, are combined together, so they are referred to as a TCP/IP. It creates a virtual connection between the source and the destination.

To facilitate the routing of packets, the TCP/IP protocol uses a 32-bit logical addresses known as IPv4 (Internet Protocol version 4).

#### Types of IP Addresses

There are two types of Internet Protocol addresses:

- Internet Protocol version 4 (IPv4):* Currently used Internet Protocol version
- Internet Protocol version 6 (IPv6):* It is an ongoing replacement for IPv4. It contains some significant improvements and new features. In this section, IPv4 will be covered in detail.

#### IP addressing scheme

There are two types of addressing schemes which are: *classful address* and *classless address*.

#### Classful addressing

This was the first addressing system implemented as part of the Internet Protocol. In 1981, the Classful addressing network architecture was first used on the Internet. The Classful addressing system was superseded by a Classless addressing scheme with the

introduction of Classless Inter-Domain Routing (CIDR) in 1993.

Classful addressing is a technique that divides the IP addresses into five categories.

### **Classful IPv4 address**

The identifier used in the network layer to identify each device connected to the Internet is the Internet address or IP address. An IPv4 address is a 32-bit address that uniquely and universally defines the connection of a device (for example, a computer or a router) to the Internet. It contains no functions for end-to-end message reliability or flow control. IP makes the ‘best effort’ to forward packets to the next destination but does not guarantee delivery because it is connectionless.

### **IP Address categories**

The IP address can be classified into two categories:

(a) *Public address*: This address is considered as any valid address assigned to any user and the organization that is responsible for registering IP ranges called Internet Service Providers (ISPs), and this address will be unique.

(b) *Private address*: this address is considered as any number or address assigned to a device on a private TCP/IP Local Area Network that is accessible only within the Local Area Network.

IPv4 is a 32-bit address which means that the maximum number of addresses that can be supported can be calculated from this formula:

Number of addresses =  $2^n$ ,

where by  $n$ =number of host bits.

$n=32$ , therefore

Number of addresses

$$= 2^{32} = 4,294,967,296$$

The IPv4 produces 4,294,967,296 addresses, which is not enough for each device connected to the internet on a planet. However, various techniques were invented, such as variable-length mask, network address translation, port address translation, classes, and inter-domain translation, to conserve the bandwidth of IP addresses and slow down the depletion of an IP address. In these techniques, public IP is converted into a private IP which the user having public IP can also use the internet. But still, this was not so efficient, giving rise to the development of the next generation (current generation) of IP addresses (IPv6).

### **Internet Protocol version 6 (IPv6)**

IPv6 is compatible with all other Internet protocols, including TCP, UDP, and ICMP.-

### **Main IPv6 features**

IPv6 is the next generation of IP addresses. The main difference between IPv4 and IPv6 is the address size of IP addresses. IPv4 is a 32-bit address, whereas IPv6 is a 128-bit hexadecimal address. IPv6 provides a large address space and contains a simple header compared to IPv4. It provides transition strategies that convert IPv4 into IPv6, and these strategies are as follows:

*Dual stacking*: It allows the user to have both versions, i.e., IPv4 and IPv6, on the same device.

**Tunnelling:** In this approach, all users who have IPv6 can communicate with an IPv4 network to reach IPv6.

**Network Address Translation:** The translation allows communication between the hosts having a different version of the IP.

Number of addresses =  $2^n$ , where by n = number of host bits.

N = 128, therefore,

Number addresses =  $2^{(128)} = 340,282,366,920,938,463,463,374,607,431,768,211,456$

This hexadecimal address contains both numbers and alphabets. Due to the usage of both the numbers and alphabets, IPv6 can produce over 340 undecillion ( $3.4 \times 10^{38}$ ) addresses.

This 340,282,366,920,938,463,463,374, 607,431,768,211,456 can be read as (340 undecillion, 282 decillion, 366 nonillion, 920 octillion, 938 septillions, 463 sextillions, 463 quintillions, 374 quadrillions, 607 trillion, 431 billion, 768 million, 211 thousand and 456.

#### Parts of classful IPv4 address

Each network interface connected to the Internet has a unique address consisting of two parts:

- The Network address:* This is the network's address within the Internet (used by gateways for routing IP packets between networks).
- The Host address:* This is the computer's address within the network (used for delivering packets to a particular network interface within the network).

#### Classful IPv4 address format

The 32-bit IP address is separated into four 8-bit octets, allowing each octet to have a value ranging from 0 to 255.

Furthermore, the IP address is logically separated into two distinct components, the network ID and the host ID. The network ID is used to identify the subnet upon which the host resides. The host ID is used to determine the host within the given subnet. Figure 3.38(a) IPv4 address format.



Format Of an IP Address

Figure 3.38(a): IPv4 address format

Each IP address consists of four sets of eight bits, each called octet, which gives out 32 bits in total. An example of the IPv4 address format is shown in Figure 3.38(b).

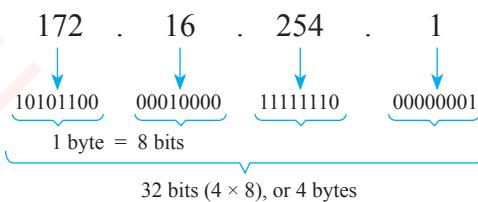


Figure 3.38(b): Example of IPv4 address format

IP addresses can be displayed in three typical formats:

**Binary notation:** This is the format systems on the network used to process the address. An example of binary notation is **11000000.10101000.00000001.01100100**

**Hexadecimal notation:** This is the format typically used when identifying IPv6 addresses. An example of

hexadecimal notation of an IPv4 address is **C0.A8.01.64**

*Dotted-decimal notation:* This is the format that is typically used for displaying the IP address in a human-readable format. An example of dotted-decimal notation is **192.168.1.100**

### Classful IPv4 Address classes

Different networks have different sizes. There are many small and large networks. To provide efficient use of 32-bit address space, IPv4 is used to define several address classes and their associated address formats, which are:

**Class A:** Allows 128 networks, 16 million hosts each

The IP address starts from 1.0.0.0 to 127.255.255.255, and the mask address is **255.0.0.0**

**Class B:** Allows 16,382 networks, 65,534 hosts each

The IP address starts from 128.0.0.0 to 191.255.255.255, and the mask address is **255.255.0.0**

**Class C:** Allows 2 million networks, 254 hosts each

The IP address starts from 192.0.0.0 to 223.255.255.255, and the mask address is 255.255.255.0

**Class D:** Multicast networks, the IP address begins from 224.0.0.0 to 239.255.255.255.

**Class E:** Reserved for future use. From 240 to 255, the 255.255.255.255 broadcasts to all the subnets.

One of the benefits of classful addresses is that they provide a hierarchy to the network through the use of the network ID. This translates into an efficient routing environment because it is easy for a router to determine what networks can be grouped and treated as a single routing entry. Table 3.9 shows the five different IP address classes.

**Table 3. 9:** Five IP address classes

| Class                     | First Octet decimal (Range) | First Octet binary (range) | IP Range                  | Subnet Mask   | Hosts per network ID | No. of networks |
|---------------------------|-----------------------------|----------------------------|---------------------------|---------------|----------------------|-----------------|
| Class A                   | 0 – 27                      | 0XXXXXXXX                  | 0.0.0.0–27.255.255.255    | 255.0.0.0     | $2^{24} - 2$         | $2^7$           |
| Class B                   | 128-191                     | 10XXXXXX                   | 128.0.0.0–191.255.255.255 | 255.255.0.0   | $2^{16} - 2$         | $2^{14}$        |
| Class C                   | 192- 223                    | 110XXXXX                   | 192.0.0.0–223.255.255.255 | 255.255.255.0 | $2^8 - 2$            | $2^{21}$        |
| Class D<br>(Multicast)    | 224 - 239                   | 1110XXXX                   | 224.0.0.0–239.255.255.255 |               |                      |                 |
| Class E<br>(Experimental) | 240-255                     | 1111XXXX                   | 240.0.0.0–255.255.255.255 |               |                      |                 |

### **Strategies to conserve addresses**

Several strategies have been developed and implemented to help the Internet community manage IP addresses well. These strategies help to reduce the load on Internet routers and help administrators use globally unique IP addresses more efficiently. There are two common strategies, which are:

- (a) Private Addressing
- (b) Classless Inter-Domain Routing (CIDR)

#### **Private address**

If the internetwork is limited to one organization, the IP addresses must be unique within that organization. Only networks that interface with public networks, such as the Internet, need public addresses. Using public addresses on the outside and private addresses for inside networks is very effective.

According to RFC1918, there are three ranges of IP addresses as private:

In class A:

10.0.0.0 through 10.255.255.255

In class B:

172.16.0.0 through 172.31.255.255

In class C:

192.168.0.0 through 192.168.255.255

There are two ways to convert a private address to a public address:

- (a) *Network Address Translation (NAT)*

This technique converts the private address to a public address, and the NAT allows us to access the internet and get services. The basic idea is that the technique uses a pool of public addresses and assigns one public address for each private address. Thus, this way is inefficient because there are costs and

delays associated with this operation.

#### **(b) Port Address Translation (PAT)**

Port Address Translation (PAT) is another technique to convert a private address to a public one. During PAT, each computer on LAN is translated to the same IP address (public) but with a different port number assignment. This way is much better than the NAT because we can use one public address to translate any private address, saving costs. Table 3.10 shows the process of the PAT.

**Table 3.10:** Example of PAT process

| Private Address | Port Address | Public Address | Port Number |
|-----------------|--------------|----------------|-------------|
| 192.68.1.3      | 50133        | 200.0.1.2      | 1           |
| 192.68.1.5      | 63887        | 200.0.1.2      | 2           |

The port address will be any random number in the allowed range the device creates when it wants to access the internet.

The packet will contain the port number assigned to the device that wants to access the internet, and through this port number, the router, when it gets the response message, will translate it and map this message to its private address on the port number.

#### **Addressing without subnet**

If we have a class B with a Flat Network, the number of hosts will be more than  $2^{16}=65,536$ . So, the problem comes here as managing this network with this number of hosts is too tricky, and this network's performance will decrease because of the heavy load. In other words, any single broadcast can slow down the network.

Therefore, the solution is Subnetting. Subnetting means dividing or separating the single network into multiple networks that can reduce the loading from one network.

The advantages of using subnetting are:

- (a) Reduces the traffic and increases the performance
- (b) The smaller network can be easier to manage

### Classful address subnetting

As the number of distinct local networks grows, managing them becomes a serious challenge. Every time a new network is installed, the system administrator must contact NIC to get a new network number.

The solution to the problem is to allow a network to be split into several independent parts for internal use but still act like a single network to the outside world. In the Internet literature, these parts are called subnets.

### Subnet masks

A mask is a 32-bit binary number expressed in dotted decimal notation. By default, a mask contains two fields, the network field and the host field. These correspond to the network number and the locally administered part of the network address. When an administrator subnets, they adjust the way they view the IP address. The default masks for classful addressing are shown in Table 3.11.

**Table 3.11:** Default masks for classful addressing

| Address class | Bits used for subnet mask |          |          |          | Dotted decimal notation |
|---------------|---------------------------|----------|----------|----------|-------------------------|
| Class A       | 11111111                  | 00000000 | 00000000 | 00000000 | 255.0.0.0               |
| Class B       | 11111111                  | 11111111 | 00000000 | 00000000 | 255.255.0.0             |
| Class C       | 11111111                  | 11111111 | 11111111 | 00000000 | 255.255.255.0           |

1 Octet      2 Octet      3 Octet      4 Octet

Routers and hosts still assume class subnet masks by default as shown in Table 3.12.

**Table 3.12:** Default subnet mask

| Class   | Subnet mask prefix | The subnet mask in dot notation |
|---------|--------------------|---------------------------------|
| Class A | /8                 | 255.0.0.0                       |
| Class B | /16                | 255.255.0.0                     |
| Class C | /24                | 255.255.255.0                   |

Three important things that should be taken into consideration when we are thinking about subnetting are network address, broadcast address, and host addresses

Moreover, to find the number of hosts per subnet, we can use the formula  $2^n - 2$ , where (n) is the number of unmasked bits (0's).

For example, in 11000000, the number of zeros gives us  $2^6 - 2 = 62$  hosts. In this example, there are 62 hosts per subnet and we subtract because the first IP address is reserved for the network address, and the last one is for the network broadcast.

When we want to find the number of networks, we can use this formula  $2^y$

Where, y represents the number of masked bits (1's). For example, in 11000000, the number of ones gives us = 4

### **Reserved and restricted addresses**

Certain addresses cannot be assigned to an individual device in any subnet because they have a special purpose. The subnet address is the first address in a range that identifies the subnet. In the range, the broadcast address is the last; all hosts on the subnet receive traffic if anything is sent to it.

For example, if a subnet address is 192.50.10.0 with a mask of 255.255.255.0, then the subnet address will be 172.31.9.0, and the broadcast address 192.5010.255.

### **Classless Inter-Domain Routing**

Classless Inter-Domain Routing (CIDR) is also called classless addressing. CIDR is an addressing scheme that aids in the more efficient allocation of IP addresses. This technique assigns a block of IP addresses based on specified conditions when the user demands a specific amount of IP addresses. This block is known as a “CIDR block”, containing the necessary number of IP addresses.

Classless addressing is concerned with the following three rules when allocating a block.

**Rule 1:** All CIDR block's IP addresses must all be contiguous.

**Rule 2:** The block size must be a power of two to be attractive. Furthermore, the block's size equals the number of IP addresses in the block.

**Rule 3:** The block's first IP address must be divisible by the block size.

This addressing scheme is used for assigning IP addresses to the device on the network without using the standard IP address classes like Class A, Class B, or Class C. In CIDR, an IP address is represented as A.B.C.D /n, where “/n” is the IP prefix or network prefix. The IP prefix identifies the number of significant bits used to identify the network.

For example, 172.168.20.70 /20 means that the first 20 bits represent the network, and the remaining 12 bits are used to identify hosts. It is basically the method that ISPs (Internet Service Providers) use to allocate several addresses to a company, a home, or a customer. They provide addresses in specific block sizes. When you receive a block of addresses from an ISP, the results you get will look something like this: 172.168.10.40/29. This tells you what your subnet mask is. The slash notation (/) means how the number bits turned on (1s).

The Class A default subnet mask is 255.0.0.0, which means that the first byte of the subnet mask is all ones (1s), or 11111111. When referring to a slash notation, you need to count all the 1s bits to figure out your mask. The 255.0.0.0 is considered a /8 because it has 8 bits which are 1s; that is, 8 bits that are turned on.

## Subnetting Class C address

In a Class C address, only 8 bits are available for defining the hosts whose subnet bits start at the left and go to the right without skipping bits. This means that the only Class C subnet masks are the following:

| Binary    | Decimal | CIDR |
|-----------|---------|------|
| 100000000 | = 128   | /25  |
| 110000000 | = 192   | /26  |
| 111000000 | = 224   | /27  |
| 111100000 | = 240   | /28  |
| 111110000 | = 248   | /29  |
| 111111000 | = 252   | /30  |

### The fast-way method for sub-netting a Class C address

Start by using the second subnet mask available with a Class C address, which borrows 2 bits for sub-netting:

$$192 = 11000000.$$

The 1s represent the subnet bits, and the 0's represent the host bits available in each subnet. 192 provides 2 bits for sub-netting and 6 bits for defining the hosts in each subnet.

**Example 1:** Given the network address 192.168.10.0 and subnet mask 255.255.255.192,

find the number of subnets.

#### Solution:

Since 192 is 2 bits on (11000000), the answer will be  $2^2 = 4$

Therefore, the number of subnets will be 4.

Find the number of hosts per subnet.

We have 6 host bits off (11000000); therefore, the number of hosts per each subnet will be:

$$2^6 - 2 = 62$$

Find the valid subnets:  $256 - 192 = 64$ ; remember, we always start from 0 and count in our block size, so our subnets are: 0, 64, 128, and 192.

**Example 2:** Your school has been granted the site address 201.70.64.0 (class C). The school need six subnets. Design the subnet for your school.

#### Solution:

Our network address is 201.20.64.0, and the default subnet mask for class C is 255.255.255.0, or you can write: 201.70.64.0/24.

The subnet mask in binary:

$$\begin{array}{l} 11111111.11111111.11111111 \\ 00000000 \end{array} = 255.255.255.0$$

The school needs six subnets, 6 is not a power of 2; therefore, you will need to go for the next number that is in the power of 2, which is  $8(2^3)$ ; so, you will need to borrow 3 bits for the network from host part.

Therefore, the new subnet mask will be:  
 $24+3=27$

The new subnet mask will be 11111111.11111111.11100000

$$= 255.255.255.224$$

Number of subnets = 8

The total number of host addresses per subnet will be  $2^5 = 32$

The number of usable host addresses per subnet will be  $2^5 - 2 = 30$ .

**Exercise 3.3**

- What is the purpose of the OSI model in the modern internet? Does it always serve that purpose? Give reasons to justify your answer.
- Suppose you have been given a network address: 172.168.10.0 and subnet mask: 255.255.255. Follow all procedures to find out the number of subnets, number of host per network and valid subnets.
- Contrast between classful and classless IP addresses.
- Describe the importance of subnetting in network .
- Explain the mechanism for data encapsulation and de-encapsulation in OSI Model.
- Describe NAT and PAT mechanism.
- Explain the OSI reference models with a neat diagram.

**Activity 3.4: Assign IP address to networked computers**

An ISP has granted a certain university a block of addresses starting with 193.164.120.0/24. Since you are an administrator, you have been tasked to distribute the addresses to three campuses as follows:

- Mwanza Campus has 50 staff's offices, each needs 120 addresses
- Dar es salaam campus has 89 staff's offices, each needs 54 addresses
- Arusha Campus has 68 staff's offices, each needs 68 addresses
  - Design the sub blocks and give the slash notation for each sub blocks.
  - Find out how many addresses will be remaining after these allocations.

**Transmission media****Concept of transmission media**

A computer network is a collection of two or more computer systems linked together. A network connection can be established using either cable or wireless media. Transmission media are used to connect nodes in any network. A computer network consists of various nodes such as; servers, networking hardware, personal computers, and other specialized or general-purpose hosts.

**Meaning of transmission media**

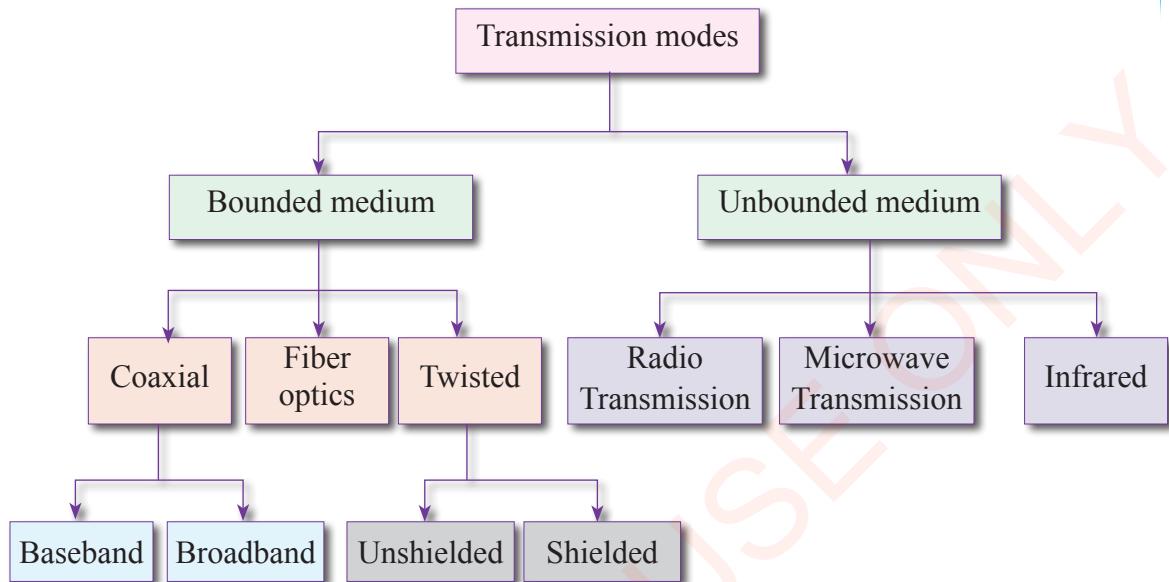
A transmission media is any communication channel that uses electromagnetic signals to transmit data from the transmitter to the receiver. The primary role is to transmit data in the form of bits across the network. Data transmission functions similarly to a physical path between the sender and the recipient. For instance, in a copper cable network, the bits are in the form of electrical signals, whereas in a fibre network, the bits are available in light pulses. The quality and characteristics of data transmission can be determined by the characteristics of the medium and signal. The properties of different transmission media are delay, bandwidth, maintenance, cost, and easy installation.

**Types of transmission media**

Transmission media are categorised into two types: Guided (wired) and unguided (wireless) transmission media. Also, guided and unguided are known as bounded and unbounded, respectively. Figure 3.39 illustrates the types of transmission media.

*Factors to be considered while selecting a transmission medium*

- (a) Transmission rate
- (b) Cost and ease of installation
- (c) Resistance to environmental conditions
- (d) Distances

**Figure 3.39: Transmission media types*****Guided transmission media***

In guided media, signals are transmitted through a physical path. The features of guide media are high speed and security. Guided media are also known as Wired or Bounded transmission media. There are three types of guided media: Twisted pair cable, Coaxial cable, and Fiber optics cable.

***Twisted pair cable***

A twisted pair cable is made up of two insulated wires twisted with each other. One wire carries the signal from the source to the destination, and the other is for ground reference. Twisting is done to avoid exterior electromagnetic interference. It is the most widely used transmission media.

***Advantages of twisted pair cable***

- (a) Installation is easy
- (b) Lightweight cable
- (c) High speed
- (d) Can carry both analog and digital data
- (e) Cheaper than other transmission media
- (f) If some part of the twisted pair cable is damaged, it does not affect the entire network

***Disadvantages of twisted pair cable***

- (a) Poor security
- (b) Not durable (easy to break)
- (c) Attenuation is high
- (d) They support only lower bandwidth

**Applications of twisted pair cables**

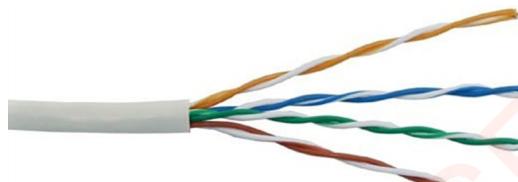
- (a) Applied in telephone lines.
- (b) Applied Local Area Networks such as small offices and data centres use twisted pair cables.

**Types of twisted pair cable**

There are two types of twisted-pair cables:

- (a) Unshielded Twisted-Pair Cable (UTP)
- (b) Shielded Twisted-Pair Cable (UTP)

**Unshielded Twisted-pair cable:** UTP cables consist of 2 or 4 pairs of twisted cables. The Cable with 2 pairs uses an RJ-11 connector, and 4 pair cables use an RJ-45 connector. RJ means Registered Jack. Figure 3.40 shows the Unshielded Twisted Pair cable (UTP).



**Figure 3.40:** Unshielded Twisted Pair cable

Unshielded twisted pair cable is widely used in telecommunications. Unshielded Twisted pair (UTP) cable is categorized as follows:

- |                              |                             |
|------------------------------|-----------------------------|
| (a) Category 1 UTP: CAT1     | (b) Category 2 UTP: CAT2    |
| (c) Category 3 UTP: CAT3     | (d) Category 4 UTP: CAT4    |
| (e) Category 5 UTP: CAT5 &5e | (f) Category 6 UTP: CAT6&6a |

The twisted pair cables and their characteristics are summarized in Table 3.13.

**Table 3.13:** Twisted Pair cable characteristics

| Cable Category | ATTRIBUTES |                    |              |            |                |                     |
|----------------|------------|--------------------|--------------|------------|----------------|---------------------|
|                | Frequency  | Max. Transm. Speed | Distance (M) | Cable type | Connector type | Applications        |
| CAT1           | -          | >=1Mbps            |              | UTP        | RJ-11          | Old telephone cable |
| CAT2           | -          | >=4Mbps            |              | UTP        |                | Token ring network  |

|       |         |                       |     |             |                             |                                           |
|-------|---------|-----------------------|-----|-------------|-----------------------------|-------------------------------------------|
| CAT3  | 16MHz   | $\geq 10\text{Mbps}$  | 100 | UTP         | RJ-45                       | Token ring and 10-BASE-T Ethernet         |
| CAT4  | 20MHz   | $\geq 20\text{Mbps}$  | 100 | UTP         | RJ-45                       | Token ring network                        |
| CAT5  | 100MHz  | $\geq 100\text{Mbps}$ | 100 | UTP         | RJ-45                       | Ethernet, token ring, Fast Ethernet       |
| CAT5e | 125MHz  | $\geq 1\text{Gbps}$   | 100 | UTP         | RJ-45                       | Ethernet, Fast Ethernet, Gigabit Ethernet |
| CAT6  | 250MHz  | $\geq 10\text{Gbps}$  | 100 | STP/<br>UTP | RJ-45                       | Gigabit Ethernet and 10Gbit Ethernet      |
| CT6a  | 500MHz  | $\geq 10\text{Gbps}$  | 100 | UTP/<br>STP | RJ-45                       | Gigabit Ethernet and 10Gbit Ethernet      |
| CAT7  | 600MHz  | $\geq 10\text{Gbps}$  | 100 | STP         | GigaGate45 (GG45) and RJ-45 | Gigabit Ethernet and 10Gbit Ethernet      |
| CAT7a | 1000MHz | $\geq 100\text{Gbps}$ | 100 | STP         | GG45 / RJ-45                | 10Gbit Ethernet                           |
| CAT8  | 2000MHz | 25/40Gbps             | 30  | STP         | GG45/ RJ-45                 | 25Gbit Ethernet and 40Gbit Ethernet       |

**Note:** TP= Twisted pair

#### *Advantages of Unshielded Twisted-Pair Cable*

- (a) It can support high data speed, which can be used in high-speed LAN implementation.
- (b) Very easy to set up and install
- (c) It is a flexible and cheap option for the physical transmission media

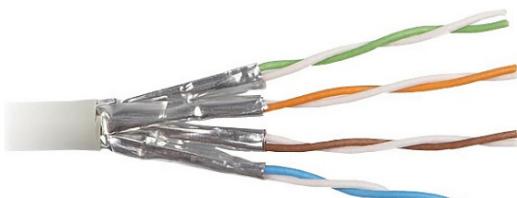
#### *Disadvantages of Unshielded Twisted-Pair Cable*

- (a) Bandwidth is limited
- (b) Noise reduction is not possible
- (c) Not ideal for long distance communications due to the attenuation.

**Shielded twisted-pair cable**

Shielded twisted pair cable contains the metal foil surrounding the twisted copper wires to avoid external interference. Shielding also eliminates crosstalk.

Shielded twisted pair cable has the same attenuation as unshielded twisted pair. It is faster than unshielded and coaxial cable. It is more expensive than a coaxial and unshielded twisted pair. Figure 3.41 shows Shielded Twisted pair.



**Figure 3.41:** Shielded Twisted pair

**Advantages of Shielded Twisted-Pair Cable**

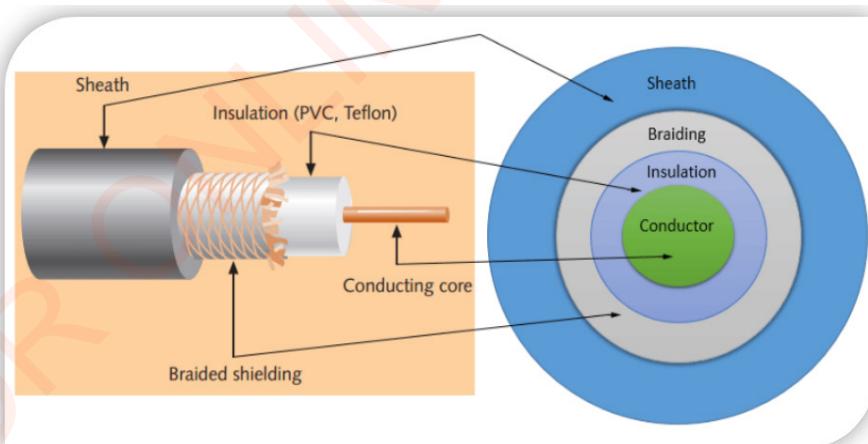
- Better performance than unshielded twisted pair cable
- Installation is easy
- It supports higher bandwidth and high speed because it's shielded.

**Disadvantages of Shielded Twisted-Pair Cable**

- It is more costly than an unshielded twisted pair cable (it is more expensive)
- It is difficult to manufacture and has heavyweight

**Coaxial cable**

Coaxial cable has a centre conductor enclosed in an insulating cover enclosed in a metal foil shield. The whole cable is covered in a plastic cover. The metal foil shield protects against the noise. As the name suggests, it consists of two parallel conductors. The inner conductor is made up of copper and the outer conductor is composed of copper mesh. An insulating cover separates these two conductors. The inner centre conductor is used for data transfer, while the outer copper mesh conductor protects against the noise. Figure 3.42 illustrates the coaxial cable.



**Figure 3.42:** Coaxial cable

### **Coaxial cable standards DO NOT DUPLICATE**

*Coaxial cables* are categorized based on their Radio Government (RG) ratings. Each RG number denotes a unique set of physical specifications, including the wire gauge of the inner conductor, the thickness and the type of the inner insulator, the construction of the shield, and the size and type of the outer casing. Each cable defined by an RG rating is adapted for a specialized function as shown in Table 3.14.

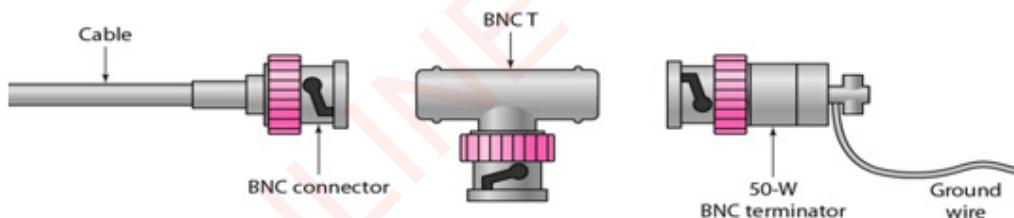
**Table 3.14:** The most common coaxial standards.

|        |               |                            |
|--------|---------------|----------------------------|
| 50-Ohm | RG-7 or RG-11 | used with thick Ethernet   |
| 50-Ohm | RG-58         | used with thin Ethernet    |
| 75-Ohm | RG-59         | used with cable television |
| 93-Ohm | RG-62         | used with ARCNET           |

### *Coaxial cable connectors*

To connect coaxial cables to devices, we need coaxial connectors. The most common type of connector used today is the Bayonet Neill-Councilman (BNC). Figure 3.43 shows three (3) popular types of these connectors: the BNC Connector, the BNC T connector and the BNC terminator.

The BNC connector is used to connect the end of the cable to the device, such as a TV set. The BNC T connector is used in Ethernet networks to branch out to a connection to a computer or other device. The BNC terminator is used at the end of the cable to prevent the reflection of the signal.



**Figure 3.43: BNC Connectors**

### *Types of coaxial cable*

- Base Band Transmission cable* is used to transmit a single signal at high speed. It is used for digital transmission.
- Broadband Transmission cable* is used for the transmission of multiple signals simultaneously. It is used for analog transmission.

### *Advantages of Coaxial cable*

- They are used in long-distance telephone lines
- Transmit digital signals at a very high rate

- (c) Data is transmitted without distortion
- (d) They can span to a longer distance at higher speeds as they have better shielding when compared to twisted pair cable
- (e) It has high-speed data transmission
- (f) Better noise reduction compared to twisted pair cable
- (g) It supports high bandwidth.

#### ***Disadvantages of coaxial cables***

- (a) They are more expensive than a twisted pair cables
- (b) Single cable failure can result in the entire network failure
- (c) Difficult to install and costly when compared to twisted pair
- (d) If the shield is imperfect, it can lead to a grounded loop.

#### ***Applications of coaxial cable***

Coaxial cable was widely used in analog telephone networks, where a single coaxial network could carry 10,000 voice signals. Cable TV networks also use coaxial cables. In the traditional cable TV network, the entire network uses coaxial cables. Cable TV uses RG-59 coaxial cable.

In traditional settings, coaxial cable it is used in the Ethernet LAN. Due to its high bandwidth, consequence and high data rate, coaxial cable was chosen for

digital transmission in early Ethernet LANs.

The 10Base-2 or Thin Ethernet uses RG-58 coaxial cable with BNC connectors to transmit data at 10Mbps with a range of 185M.

#### ***Fibre optic cable***

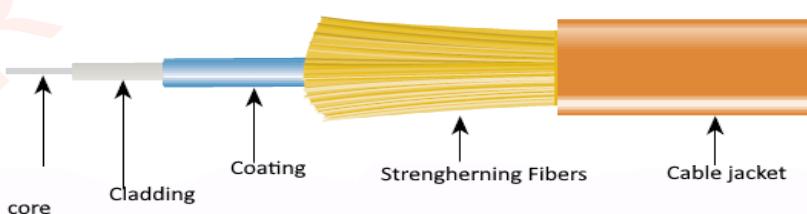
*Fibre optics cable* is made of glass and data transmission is based on the concept of reflection of light through glass. In fibre optic cable, the optical fibres are coated in plastic. These optical fibres send the data by pulses of light. The plastic coating protects the cable from temperature, electromagnetic inference, and other wiring issues. Figure 3.44 shows the fibre optic cable components.

#### ***Components of a fibre optic cable***

*Core*: The core of the fibre optic cable contains optical fibres and is responsible for data transmission in the form of light pulses.

*Cladding*: The core is enclosed in a layer of glass, which is called *cladding*. The main use of cladding is to reflect the light so that the light can be transmitted from the source to the destination through reflection.

*Jacket*: The jacket is the outer layer of fibre optic cable made up of plastic, and the whole purpose of the jacket is to protect the cable.



**Figure 3.44: Fiber optics cable components**

***Fibre optic cable propagation modes***

**Multimode:** Multiple light sources move through different paths inside the coaxial cable. Multiple data transmission is done simultaneously.

**Single-mode:** Light moves in a single path close to the horizontal of the cable.

***Advantages of fibre optic cable***

- (a) The fibre optic cable supports high bandwidth compared to the copper cables. Thus, it transfers more data compared to the copper cables.
- (b) Since the data is transmitted in the form of light, the transmission speed is very high.
- (c) The fibre optic cable can be used for more extensive distance data transmission than copper wires cable.
- (d) It is more reliable than copper cables as the core is protected against temperature and electromagnetic disturbances.
- (e) It is durable
- (f) It is lightweight
- (g) It is resistant to corrosive materials
- (h) Fibres do not leak light and are quite difficult to tap
- (i) There is no cross-talk problem in optical fibres
- (j) Optical fibres cost lower than cables with metallic conductors

***Disadvantages of fibre optic cable***

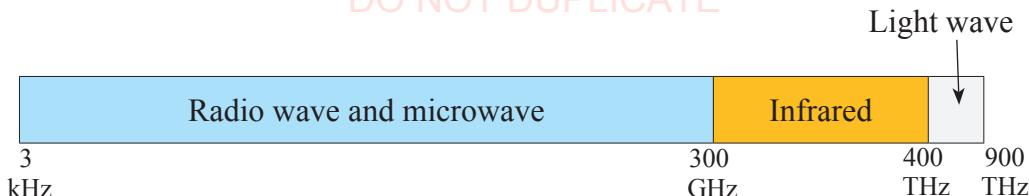
- (a) It requires skilled people to administer, install and maintain.
- (b) Optical fibres are unidirectional for two-way communication; so two fibres are required.
- (c) It is too expensive to deploy.
- (d) Fibre optics cables are fragile, i.e., they are easier to break than wires.

***Applications of Fiber Optic Cable***

- (a) Often found in backbone networks because its wide bandwidth is cost-effective.
- (b) Some cable TV companies use a combination of optical fibre and coaxial cable, thus creating a hybrid network.
- (c) Optical fibres are now used in telephone systems
- (d) Used in the Local Area Networks

***Unguided transmission media***

Unguided media does not use a physical medium to transmit the data from source to destination. This type of communication is often referred to as wireless communication. Unguided media uses air as a transmission medium to transfer data. Thus, it is called wireless transmission media. The unguided media uses an antenna for transmitting and receiving the data in the form of electromagnetic waves. Figure 3.45 shows the part of the electromagnetic spectrum, ranging from 3 kHz to 900 THz, used for wireless communication.

**Figure 3.45:** Electromagnetic spectrum**Wireless signal propagation modes**

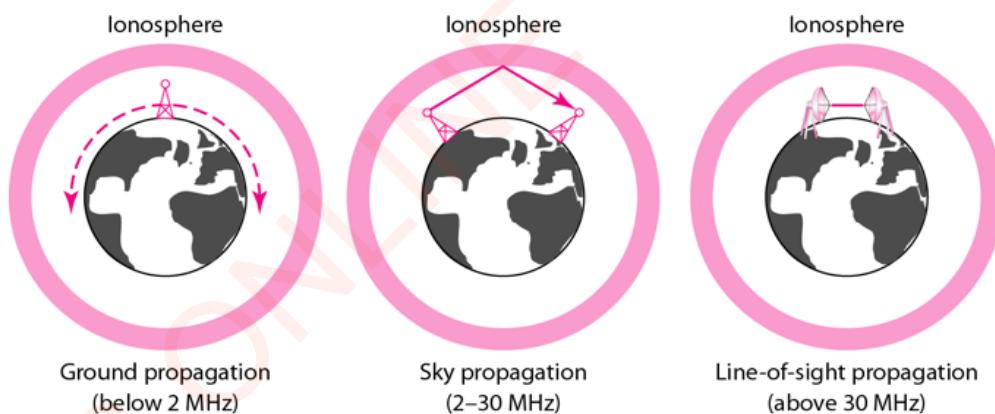
Wireless signals can travel from the source to the destination using three types of propagation modes: Ground propagation, sky propagation and line-of-sight propagation. Figure 3.46 shows wireless propagation modes.

(a) *Ground Propagation*: In this mode, radio waves travel through the lowest portion of the atmosphere, hugging the Earth. These low-frequency signals originate in all directions from the

transmitting antenna and follow the planet's curvature.

(b) *Sky Propagation*: In this mode, higher-frequency radio waves radiate upward into the ionosphere, reflecting back to the earth. This type of transmission allows for greater distances with lower output power.

(c) *Line-of-sight Propagation*: in this type, very high-frequency signals are transmitted directly from antenna to antenna in straight lines.

**Figure 3.46:** Wireless signal propagation modes

Types of wireless transmission media DUE TO IOT

- Microwave transmission
- Radio transmission
- Infrared transmission

### ***Microwave transmission***

Electromagnetic waves having frequencies between 1 and 300 GHz are called microwaves. Microwaves are unidirectional. When an antenna transmits microwaves, they can be narrowly focused. This means that the sending and receiving antennas need to be aligned. The unidirectional property has an obvious advantage. A pair of antennas can be aligned without interfering with another pair of aligned antennas.

### *Characteristics of microwaves propagation*

- Microwave propagation is line-of-sight since the towers with the mounted antennas need to be in direct sight of each other, and towers that are far apart needed to be very tall.
- Very high-frequency microwaves cannot penetrate walls. This characteristic can be a disadvantage if receivers are inside the buildings.

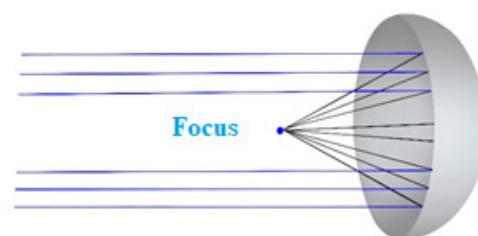
- The microwave band is relatively wide, almost 299 GHz. Therefore, wider sub-bands can be assigned, and a high data rate is possible.
- Use of certain portions of the band requires permission from authorities.

### *Antenna for Microwaves*

Microwaves need unidirectional antennas that send out signals in one direction. Two types of antennas are used for microwave communications: Parabolic Dish and Horn.

*A parabolic antenna:* It works as a funnel, catching a wide range of waves and directing them to a common point. In this way, more of the signal is recovered than would be possible with a single-point receiver.

*A horn antenna:* It looks like a huge scoop. Outgoing transmissions are broadcast up a stem and deflected outward in a series of narrow parallel beams by the curved head. Received transmissions are collected by the scooped shape of the horn, like the parabolic dish, and are deflected down into the stem as shown in Figures 3.47(a) and (b).



**Figure 3.47: (a) Parabolic antenna (b) Horn antenna**

## *Applications of Microwaves*

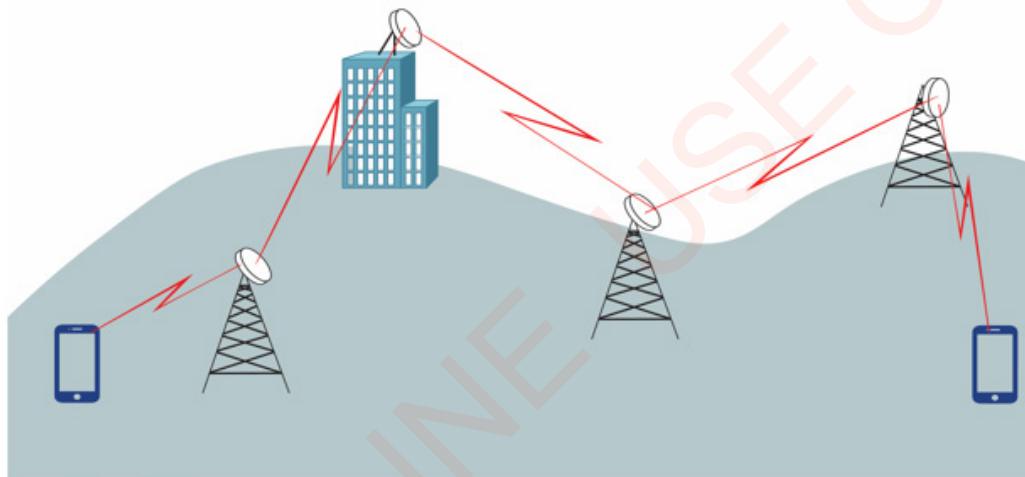
**DO NOT DUPLICATE**

Due to their unidirectional properties, microwaves are very useful when unicast (one-to-one) communication is needed between the sender and the receiver. They are used in cellular phones, satellite networks, and wireless LANs. There are two types of microwaves transmission;

- (i) Terrestrial Microwave Transmission
- (ii) Satellite Microwave Transmission

### *Terrestrial type microwave transmission*

In this type of microwave transmission, both the sender and receiver antennas are in direct sight of each other. Repeaters can be installed with each antenna to increase the distance served by terrestrial microwaves. The signal received by an antenna can be converted into transmittable form and relayed to the next antenna. It is an example of telephone systems all over the world. Figure 3.48 illustrates terrestrial microwave transmission.



**Figure 3.48: Terrestrial microwave transmission.**

#### *Features of terrestrial type microwave transmission*

**Frequency range:** The frequency range of terrestrial microwave ranges from 4 to 6GHz.

**Bandwidth:** It supports bandwidth from 1 to 10 Mbps.

**Cost:** If used in short-distance communication, the implementation cost is cheap as we do not need relay towers. However, when terrestrial type microwave transmission is used for long distances, the implementation cost goes high as we need relay towers to extend the signals.

**Attenuation:** Attenuation means loss of signal. In terrestrial type microwave transmission, attenuation mainly depends on the frequency and signal strength.

**Implementation:** Implementation is difficult as the tower needs to be in the line of sight and the signal requirement makes it even more challenging to implement.

#### **Advantages of terrestrial type microwave transmission:**

- (a) It is cheaper than cable transmission
- (b) Possible to implement in areas where cable transmission is difficult to implement, such as hill areas.

#### **Disadvantages of terrestrial type microwave transmission:**

- (a) Not secure, susceptible to eavesdropping,
- (b) Weather conditions can affect the transmission,
- (c) Limited bandwidth,

#### **Satellite microwave transmission**

This is a microwave relay station that is placed in outer space. The satellites are launched either by rockets or space shuttles carrying them.

These are positioned at 36000Km above the equator with an orbit speed that exactly matches the earth's rotation speed. As the satellite is positioned in a geo-synchronous orbit, it is stationary relative to the earth and always stays over the same point on the ground. This is usually done to allow ground stations to aim the antenna at a fixed point in the sky.

This type of communication uses the satellite antenna for communication. The sender antenna sends the signal to the satellite antenna that revolves around the earth at a known height. The satellite antenna then amplifies the received signal and sends it to the

receiver antenna. Figure 3.49 shows satellite microwave transmission.



**Figure 3.49: Satellite microwave transmission**

#### **Features of satellite microwave transmission**

- (i) Bandwidth capacity depends on the frequency used.
- (ii) Satellite microwave deployment for orbiting satellites is difficult.

#### **Advantages of satellite type transmission:**

- (a) The transmitting station can receive back its transmission and check whether the satellite has transmitted information correctly.
- (b) The transmission can be done over long distances
- (c) Unlike terrestrial transmission, where the implementation cost goes higher based on the transmission distance, satellite communication is unaffected by the data transmission distance.
- (d) Easy to install compared to terrestrial type

#### **Disadvantages of satellite type transmission**

- (a) Installation of the satellite is extremely tough.

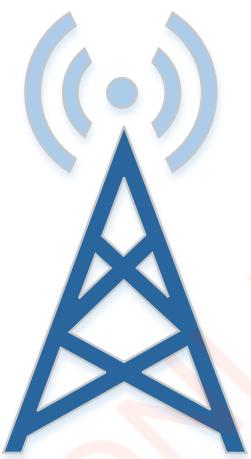
- (b) Installation cost is high, and development requires more time.

### **Radio transmission**

Radio waves are very low-frequency electromagnetic waves that are transmitted in all directions within the range. Radio transmission frequency ranges from 3KHz to 1GHz. In radio transmissions, the antennas need neither be aligned nor required to be set in a particular manner to send and receive signals. Since radio waves are omnidirectional (in all directions), isolating communication inside a building is impossible.

#### *Antenna for radio waves*

Radio waves use omnidirectional antennas that send signals in all directions, as illustrated in Figure 3.50.



**Figure 3.50. Applications of Radio Waves**

The omnidirectional characteristics of radio waves make them useful for multicasting in which there is one sender but many receivers. Amplitude Modulation (AM) and Frequency Modulation (FM) radio, television, maritime radio, cordless phones, and paging are examples of multicasting.

#### *Advantages of radio transmission*

- Radio waves are mainly used for Wide Area Networks (WAN), for example, mobile networks.
- Suitable for long distance communications.
- Signals can penetrate walls.
- Higher data transmission rate compared to other transmission mediums.

#### *Disadvantages of radio transmission*

- Waves are omnidirectional, so they face interference from other signals of the same frequency. Therefore, there can be disturbances.
- Not possible to isolate the communication inside the building

### **Infrared transmission**

Infrared waves are the highest frequency waves, and their frequency ranges from 300GHz to 400 THz.

#### *Characteristics of infrared transmission*

- Suitable for short-distance communication.
- Bandwidth is high, so the data transmission rate is high compared to other mediums.
- These waves cannot penetrate walls; thus, they are ideal for isolated communications.

#### *Advantages of infrared transmission*

- Secure
- High speed
- High-frequency signals
- High data transfer rate
- Low interference

#### *Disadvantages of infrared transmission:*

- Sun rays interfere with infrared rays, so they are not ideal for outdoor communication.

- (b) Suitable for short-distance communication only.
- (c) It cannot penetrate walls, so difficult to establish communication between two different walls.

### ***Infrared Applications.***

The infrared band, almost 400 THz, has an excellent potential for data transmission. Such a wide bandwidth can be used to transmit digital data with a very high data rate.

The Infrared Data Association (IrDA), an association sponsoring infrared waves, has established standards for using these signals for communication between devices such as keyboards, mice, PCs, and printers.

Infrared signals can be used for short-range communication in a closed area using line-of-sight propagation.

Explain to him how would he make use of this antenna to suit his purpose. Give the consequences that Makame will encounter for using these antennas in transmitting data over a long distance.

2. Describe Infrared and Radio waves transmission mechanism
3. Explain how to set terrestrial antenna for microwave transmission.
4. With the aid of a diagram, explain how the horn antenna works.
5. How parabolic antenna differ from horn antenna?
6. Describe the types of propagation modes.
7. Describe the components of fibre optic cable.

### **Exercise 3.4**

1. Makame wants to make business on data communication. He wants to establish a communication centre that will enable customers to send data over both short and long distance. How would you describe the classification of transmission media to him?
  - (a) What are factors would you advise him to consider in the selection of transmission medium?
  - (b) Makame has already bought the terrestrial antenna that he want to install for long distance communication.

### **Wireless network**

#### **Concepts of wireless network**

Wireless means the media made up of electromagnetic waves (EM Waves) or infrared waves without a wire. Antennas or sensors will be present on all wireless devices. Cellular phones, wireless sensors, TV remotes, satellite dish receivers, and laptops with WLAN cards are all examples of wireless devices. A wireless network uses radio frequency waves rather than wires for data or voice communication. Installation of wireless networks, particularly WLANs, can bring convenience and accessibility, allowing businesses to gain maximum mobility and flexibility around the office or at home.

## Wireless Local Area Network standards

Wi-Fi is the industry-standard wireless local area network (WLAN) technology for linking computers and other electronic devices to the Internet. It is a wireless variant of a wired Ethernet network that is frequently used in conjunction with it. Wi-Fi is a type of wireless networking that uses radio frequencies to send and receive data. It allows users to connect to the Internet at high speeds without the necessity of cables. Wi-Fi stands for “wireless fidelity” and is a phrase that is often used to refer to wireless networking technologies. A wireless router is used to connect to the internet. When you connect to Wi-Fi, you’re connecting to a wireless router that connects your Wi-Fi-enabled devices

### Wi-Fi operation mechanism

The IEEE 802.11 standard defines the protocols that allow existing Wi-Fi-enabled wireless devices such as wireless routers and access points to communicate. Wireless access points support different IEEE standards. Each standard results from a series of amendments that have been ratified over time. The standards operate at different frequencies, have different bandwidths, and support varied channel counts.

### Types and characteristics of various IEEE 802.11 wireless standards

Table 3.15 illustrates the type and characteristics of various IEEE 802.11 wireless standards.

**Table 3.15:** The 802.11 wireless standards

| IEEE Standard | Frequency/Medium | Speed     | Topology             | Transmission Range                                                      | Access method |
|---------------|------------------|-----------|----------------------|-------------------------------------------------------------------------|---------------|
| 802.11        | 2.4GHz RF        | 1-2Mbps   | Adhoc/infrastructure | 20 feet indoors                                                         | CSMA/CA       |
| 802.11a       | 5GHz             | >=54Mbps  | Adhoc/infrastructure | 25-75 feet indoors, the range can be affected by building materials     | CSMA/CA       |
| 802.11b       | 2.4GHz           | >=11Mbps  | Adhoc/infrastructure | Up to 150 feet indoors, the range can be affected by building materials | CSMA/CA       |
| 802.11g       | 2.4GHz           | >=54Mbps  | Adhoc/infrastructure | Up to 150 feet indoors, the range can be affected by building materials | CSMA/CA       |
| 802.11n       | 2.4GHz/5GHz      | >=600Mbps | Adhoc/infrastructure | 175+feet indoors, the range can be affected by building materials       | CSMA/CA       |

## Types of wireless network

Table 3.16 provides an overview of the different types of wireless networks.

**Table 3.16:** Wireless network types

| S/N | Network types                                                                      | Geographical range                                           | Usage                                                                                                                                      | Standards                                        |
|-----|------------------------------------------------------------------------------------|--------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------|
| 1.  | Wireless Personal Area Network (WPAN)                                              | Within reach of individual                                   | Alternative or replacement to cables for peripherals                                                                                       | Bluetooth, Zigbee, NFC                           |
| 2.  | Wireless Local Area Network (WLAN)                                                 | Within the building or campus                                | Wireless extension of wired network                                                                                                        | IEEE82.11 (Wi-Fi)                                |
| 3.  | Wireless ad hoc network (wireless mesh network or mobile ad hoc network, or MANET) | 100M but can be extended by multi-hop communication of nodes | Variety of applications where central nodes cannot rely upon, i.e., field operation, surveillance networks, home and street light networks | Not restricted to any one technology or protocol |
| 4.  | Wireless Metropolitan Area Network (WMAN)                                          | Citywide                                                     | Allows several WLANs to interconnect to cover a metropolitan area and provide a connection to a WAN                                        | IEEE 802.116 (WiMAX)                             |
| 5.  | Wireless Wide Area Network (Wireless MAN or WWAN)                                  | Regional, National or Global                                 | Delivered to smartphones and other handheld devices                                                                                        | GSM/UMTS, CDMA one/ CDMA2000 and WiMAX           |
| 6.  | Cellular network                                                                   | Regional, National or Global                                 | Voice and other cellular networks                                                                                                          | Cellular (UMTS, LTE, 0G-5G etc.)                 |

## Basic components of wireless network

**End device:** This can include, but is not limited to, devices such as, PCs, Printers, Servers, etc.

**Wireless network card/wireless adapter:** Wireless adapter is categorized into two forms: USB or PCI card. It can connect the specific computing device to WLAN and receive and transmit radio waves. Modern devices have an inbuilt wireless adapter.

**Wireless Access Point (WAP or AP):** Is the wireless network device that provides a wireless connection to the remaining wired LAN. Some of the APs devices look very similar to the wireless router. AP receives and transmits data to all wireless devices in a particular area. It can manage multiple connections between different devices working at once.

**Wireless Routers:** Most modern wireless router integrates WAP, Ethernet switch, and router into a single device. Wireless router permits wired and wireless to connect to the internet via a cable (ADSL or DSL modem)

**Wireless Bridge:** Operates to either improve Wi-Fi network coverage or extend it. The primary purpose of the device is to receive the existing Wi-Fi signal, relay requests, and respond back and forth between the devices and the main Wi-Fi router/AP. The repeater can be used to double the WLAN range. Wireless bridges are available in the high-speed, long-range outdoor link to use between buildings.

**Wireless LAN controller:** Wireless LAN network controllers Figure 3.51 is used on the network when many access points (AP) are required. As a centralized device in the network, the wireless LAN controller is usually located at the data centre, to which all the wireless APs on the network are directly or indirectly connected. It provides centralized Wi-Fi management functions that manage all campus access points. The wireless controller offers advanced features and management capabilities such as centralized authentication, access control, the ability to restrict bandwidth by user or group, quality of service (QoS), and traffic prioritization.



Figure 3.51: Wireless Network controller

## Wireless LAN controller operation

The wireless access controller takes the bandwidth produced from a router and stretches it to fit the network's needs. Similar to an amplifier in a stereo system, the wireless controller allows devices from farther distances to connect. Besides, it enables the network administrators to check all the network data and detect rogue access points and recent traps generated by the access points.

### Benefits of a wireless LAN controller

*Secured wired and wireless network:* Controlling wireless users' access privileges using a variety of deterministic criteria (including authentication method, device type and the application requested) to provide differentiated access for maintaining security.

*Centralized and flexible network management:* Centralised wireless controller provides flexibility for deployment, which will reduce the overall budget, planning tools, and time spent to organize a wireless network in the business. They enable the centralised monitoring of the entire wireless infrastructure which decreases the total cost of ownership and unification of wired and wireless access, a future-proofing investment for upgrades.

### Simplified network maintenance:

Wireless LAN controllers eliminate site surveys by including intelligent RF planning software. The self-configuring and self-healing wireless network is better for management and troubleshooting. It can accurately locate and identify each user. Due to the RF nature, the wireless access

controller can easily detect interference between nearby APs and automatically re-configure their power and channel settings. If an AP goes down, it can instruct nearby APs to increase their power levels to fill the coverage gap.

### Wireless repeater

A *Wi-Fi repeater* or extender is used to extend the coverage area of your Wi-Fi network. It works by receiving your existing Wi-Fi signal, amplifying it, and transmitting the boosted signal. With a Wi-Fi repeater, you can effectively double the coverage area of your Wi-Fi network - reaching far corners of your home or office, different floors, or even extending coverage to your yard. Figure 3.52 shows the wireless repeater.



Figure 3.52: Wireless repeater

### Wired and wireless computer networks comparison

There are differences between wired and wireless computer networks. Table 3.17 describes the differences between wired and wireless networks.

**Table 3.17:** Differences between wired and wireless network

| S.No | <b>Wired network</b>                                                                                             | <b>Wireless network</b>                                                                                                                                               |
|------|------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1.   | A wired network employs wires to link devices to the Internet or another network, such as laptops or desktop PCs | Without the wire, “ Wireless “ means media made up of electromagnetic waves (EM Waves) or infrared waves. Antennas or sensors will be present on all wireless devices |
| 2.   | Faster transmission speed                                                                                        | Slow transmission speed                                                                                                                                               |
| 3.   | Propagation delay is low                                                                                         | Propagation delay is high                                                                                                                                             |
| 4.   | More secure & hence reliable                                                                                     | Less secure and therefore less reliable                                                                                                                               |
| 5.   | Less expensive                                                                                                   | More expensive                                                                                                                                                        |
| 6.   | High installation and maintenance cost                                                                           | Low installation and maintenance cost                                                                                                                                 |

### Factors to consider in choosing a wired or wireless network

In order to come up with the right solution, whether to go for a wired or wireless network, you need to carry on wireless network requirement needs analysis by focusing on the followings:

- (a) Coverage requirements: Does your network cover a small or large area?
- (b) Network mobility: Assess how the user will be moving around. Will users need a reliable network in all areas?
- (c) Security: You need to evaluate the level of protection your data need against threats
- (d) Compatibility issues: Record your existing applications and systems to check whether the wireless network will be the right choice or not
- (e) Number and type of users: Identify the number of users, number, and type of devices which will be connected to the wireless network.

### Multiplexing

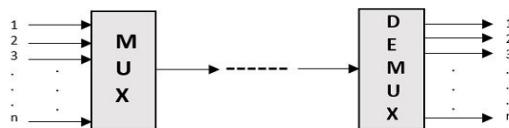
#### Concept of multiplexing

*Multiplexing* is the process of combining multiple signals into one signal over a shared medium. If the analog signals are multiplexed, then it is called *analog multiplexing*. Similarly, if the digital signals are multiplexed, then it is called *digital multiplexing*.

Multiplexing was first developed in telephony. A number of signals were combined to send through a single cable. The process of multiplexing divides a communication channel into several logical channels, allotting each one for a different message signal or a data stream to be transferred. The device that does multiplexing can be called a *Multiplexer* or MUX.

The reverse process, i.e., extracting the number of channels from one channel, which is done at the receiver, is called *de-multiplexing*. The device that does de-multiplexing can be called as *de-multiplexer* or DEMUX. In data transmission, both the multiplexer and de-

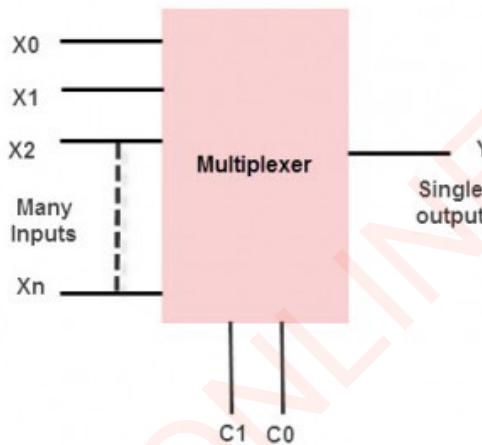
multiplexer are combinational circuits. Figure 3.53: illustrate the concept of MUX and DEMUX. Their primary use is in the field of communications.



**Figure 3.53:** Multiplexing and De-multiplexing

### Multiplexer and De-multiplexer

**Multiplexer:** The multiplexer is a device with multiple inputs and single line output. The select lines define what input is connected to the output and increase the data sent over a network at a given time. Also, it is called a data selector. Figure 3.54 shows a multiplexer.



**Figure 3.54:** Multiplexer

Multiplexers are able to accept analog and digital applications, whereby in analog applications, multiplexers contain relays and transistor switches. In contrast, in digital applications, the multiplexers are made up of

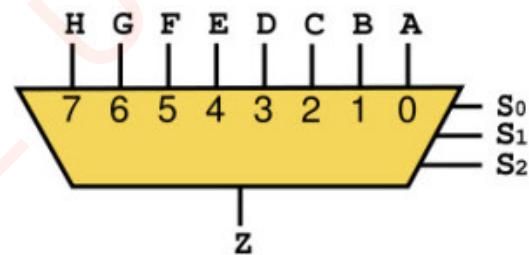
standard logic gates. When it is used for digital applications, the multiplexer is called a digital multiplexer.

### Multiplexer types

Multiplexers are classified into four types:

- 2-1 multiplexer (1 select line)
- 4-1 multiplexer (2 select lines)
- 8-1 multiplexer (3 select lines)
- 16-1 multiplexer (4 select lines)

The multiplexer is a combinational circuit with 2<sup>nd</sup> input signal lines and a single output line. For example, 4 - 1 multiplexer has 4 input data lines and 1 output line. The 8 - 1 multiplexer has 8 input data lines and 1 output line, as in Figure 3.55. The multiplexer is used for the multiplexing technique as it combines multiple signals and produces a single output signal.



**Figure 3.55:** The 8-to-1 Multiplexer with 8 input line, 1 output line and 3 selector lines

### Importance of using multiplexing

Multiplexing reduces the number of physical connections or wireless channels to transmit multiple signals by combining them and sending them over a single transmission medium. It reduces the cost of transmission services. Bandwidth is utilized effectively when using a multiplexer.

***Applications of multiplexers***

Multiplexers are used in various applications wherein multiple data must be transmitted using a single line.

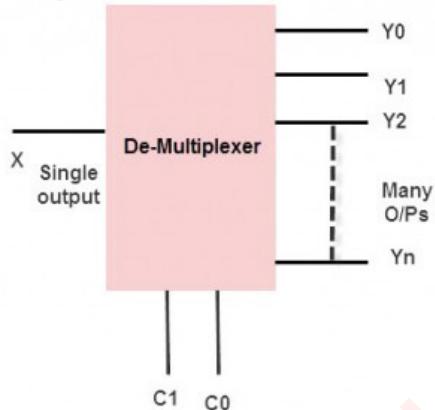
**Communication system:** A communication system has both a communication network and a transmission system. By using a multiplexer, the efficiency of the communication system can be increased by allowing the transmission of data, such as audio and video data, from different channels through single lines or cables.

**Computer memory:** Multiplexers are used in computer memory to maintain a huge amount of memory in the computers and reduce the number of copper lines required to connect the memory to other computer parts.

**Telephone network:** In telephone networks, multiple audio signals are integrated on a single line of transmission with the help of a multiplexer.

**Transmission from the computer:** The system of a satellite multiplexer is used to transmit the data signals from the computer system of a spacecraft or a satellite to the ground system by using a GSM satellite.

**De-multiplexer (DEMUX):** De-multiplexer does exactly the opposite of multiplexing. It receives a single input signal and generates 2<sup>nd</sup> output signal. The multiplexer is used at the source to combine the multiple input signals, while De-multiplexer is used at the receiving side to generate multiple signals from the received signal. Figure 3.56 shows De-multiplexer.



**Figure 3.56: De-multiplexer**

***Types of de-multiplexer***

De-multiplexers are classified into four types

- (i) 1-2 de-multiplexer (1 select line)
- (ii) 1-4 de-multiplexer (2 select lines)
- (iii) 1-8 de-multiplexer (3 select lines)
- (iv) 1-16 de-multiplexer (4 select lines)

**De-multiplexer** works on the principles of one-to-many. For example, a 1-4 De-multiplexer is used to generate 4 output signals from a single signal

***Applications of de-multiplexer***

De-multiplexers are used to connect a single source to multiple destinations. These applications include the following:

**Communication system:** Multiplexer and de-multiplexer are both used in communication systems to carry out the data transmission process. A de-multiplexer receives the output signals from the multiplexer; and, at the receiver end, it converts them back to the original form.

**Arithmetic logic unit:** The output of the arithmetic logic unit is fed as an input to

the de-multiplexer, and the output of the de-multiplexer is connected to multiple registers. The output of the ALU can be stored in multiple registers.

*Serial to parallel converter:* The serial to parallel converter is used to reform parallel data. This method gives serial data as input to the de-multiplexer at

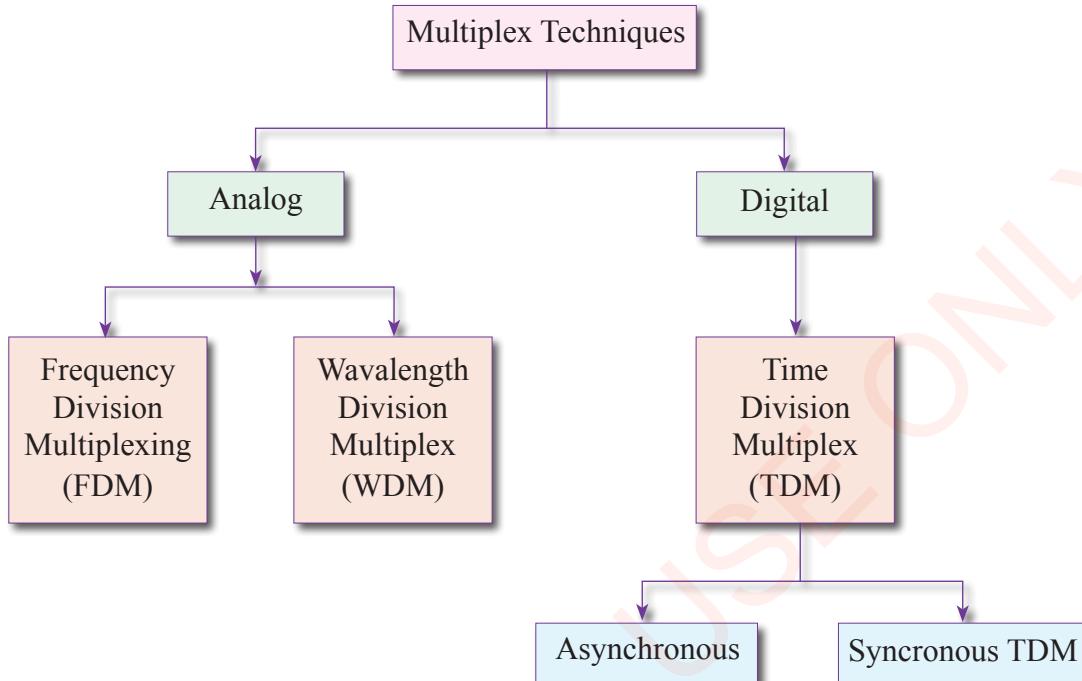
regular intervals. A counter is attached to the de-multiplexer at the control input to sense the data signal de-multiplexer's output. When all data signals are stored, the output of the de-multiplexer can be read out in parallel. Table 3.18 summarise the difference between the multiplexer and the de-multiplexer.

**Table 3. 18:** Difference between multiplexer and de-multiplexer

| Multiplexer                                                                                                                          | De-multiplexer                                                                                                     |
|--------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------|
| In multiplexers, the set of selection lines is used to control the specific input                                                    | In de-multiplexer, the output line selection can be controlled through n-selection lines bit values                |
| It uses several data inputs to generate a single output                                                                              | It uses single input that can be directed through several outputs                                                  |
| The multiplexer has several inputs and the single output                                                                             | De-multiplexer has a single input and several outputs                                                              |
| It is used in Time Division Multiplexing (TDM) at the end of the transmitter                                                         | It is used in TDM at the end of the receiver                                                                       |
| It is a digital switch                                                                                                               | It is a digital circuit                                                                                            |
| Parallel to serial conversion is used                                                                                                | Serial-to-parallel conversion is used                                                                              |
| It is a data selector                                                                                                                | It is a data distributor                                                                                           |
| It is called MUX                                                                                                                     | It is called DEMUX                                                                                                 |
| In multiplexers, control signals are used to choose the specific input that has to be sent to the output                             | De-multiplexer uses the control signal to permit us to include several outputs                                     |
| Different types of multiplexers are 4-1 MUX, 8-1 MUX, 16-1 MUX, and 32-1 MUX                                                         | Different types of de-multiplexers ar:1-4 DEMUX,1-8 DEMUX, 1-16 DEMUX, and 1-32 DEMUX                              |
| It operates on the principle of many to one                                                                                          | It operates on the principle of one-to-many                                                                        |
| It is used to improve the efficiency of the communication system using transmission data like transmission of audio as well as video | De-multiplexer gets the output signals from the Mux and changes them to the unique form at the end of the receiver |
| It doesn't use any extra gates while designing it                                                                                    | Additional gates are necessary while designing DEMUX                                                               |

## Types of multiplexing techniques

There are mainly two types of multiplexing techniques, namely analog and digital. They are further divided into Frequency Division Multiplexing (FDM), Wavelength Division Multiplexing (WDM), and Time Division Multiplexing (TDM) as it is illustrated in Figure 3.57.



**Figure 3.57: Multiplexing techniques types**

### Analog multiplexing

The signals used in analog multiplexing techniques are analog in nature. The analog signals are multiplexed according to their frequency or wavelength.

*Frequency division multiplexing:* The most used technique is FDM in analog multiplexing. This technique uses various frequencies to combine data streams, for sending them on a communication medium as a single signal.

For example, a traditional television transmitter, which sends several channels through a single cable, uses FDM.

#### Advantages of frequency division multiplexing

- It doesn't require synchronization between the sender and receiver.
- It is simple, easy to implement and a less expensive technique.
- FDM is a reliable system compared to other multiplexing systems.
- This technique is mainly used for analog signals.
- This technique allows a large number of signals to be transmitted simultaneously over a single transmission medium.

### *Disadvantages of frequency division multiplexing*

- FDM suffers from interference caused by electromagnetic signals, generally referred as cross talk.
- FDM provides less throughput. Throughput is the rate of successful message delivery over a communication channel.
- FDM doesn't utilize the full bandwidth of the transmission medium.
- FDM requires a significantly large number of modulators and filters to transmit multiple signals.
- Unlike TDM, FDM needs a carrier signal to transmit multiple signals.

*Wavelength division multiplexing:* WDM is an analog technique in which many data streams of different wavelengths are transmitted in the light spectrum. If the wavelength increases, the frequency of the signal decreases. A prism, which can turn different wavelengths into a single line can be used at the output of MUX and input of DEMUX.

### *Advantages of wavelength division multiplexing*

- Easy to implement
- Full duplex transmission is possible in WDM
- Provides better security of data
- WDM provides higher bandwidth

### *Disadvantages of wavelength division multiplexing*

- Transmitted signals cannot be very close to each other
- WDM is more expensive compared to FDM
- It is difficult to fine-tune wavelength in the WDM technique

For example, Optical fibre communications use the WDM technique to merge different wavelengths into a single light for communication.

### **Digital multiplexing**

The term digital represents discrete bits of information. Hence, the available data is in the form of frames or packets, which are discrete.

*Time-division multiplexing:* In TDM, the time frame is divided into slots. This technique is used to transmit a signal over a single communication channel, by allotting one slot for each message. TDM can be classified into Synchronous TDM and Asynchronous TDM.

*Synchronous TDM:* In synchronous TDM, the input is connected to a frame. If there are 'n' number of connections, then the frame is divided into 'n' time slots. One slot is allocated for each input line.

In this technique, the sampling rate is common for all signals, and hence the same clock input is given. The MUX allocates the same slot to each device at all times.

*Asynchronous TDM:* In asynchronous TDM, the sampling rate is different for each signal, and a common clock is not required. If the allotted device for a time slot transmits nothing and sits idle, then that slot can be allotted to another device, unlike synchronous

This type of TDM is used in asynchronous transfer mode networks.

The following are common examples of time-division multiplexing applications:

- It is used in a type of telephone line that uses Integrated Services Digital Network (ISDN) technology.

- (b) It is deployed in Public Switched Telephone Network (PSTN).
- (c) It's a significant component of a telephone system.
- (d) In a digital audio mixing system, TDM is implemented.
- (e) It is employed in a half-duplex communication system.
- (f) TDM is a technique utilized in Global Systems for Mobile Communication.

*Advantages of time-division multiplexing*

- (a) *Easy adaptation:* TDM can surely adapt to the transport of information in the same fashion as voice communication.
- (b) *Fairly high transmission speed:* It can transport messages at speeds spanning from 64 kbps to 120 Mbps.
- (c) *Absence of external impedance:* The synchronous transmission has no impedance.
- (d) *Digitization potential:* TDM is a sophisticated way to transform a simple framework into a digitized one.
- (e) *Bandwidth efficiency:* The signal is conveyed using the maximum channel bandwidth.
- (f) *No particular need for the carrier:* TDM systems do not necessitate a carrier wave or carrier signal, although FDM systems must. Carrier signal partitions the main signal as per frequency range.
- (g) *No intermodulation disruption:* TDM does not tend to suffer from Intermodulation distortions.

**D(h)** *Crosstalk is not a huge issue:* Crosstalk can be triggered by pulse overlapping; that being said, it can be minimized by employing guard time. As a consequence, there is not anything to be particularly concerned about.

- (i) *Ensures optimal performance:* Dynamic coordination is a core feature of TDM. Since TDM uses dynamic coordination, it is more efficient and reliable than FDM.
- (j) *A diverse range of application areas:* TDM allows the administrator to do operations such as faxing, voiceband data, and SMS, as well as activities such as mixed media and video calls.
- (k) *Cost-benefit analysis:* TDM brings considerable investment funds in base station equipment, space, and support, which is becoming extremely important as cell sizes shrink.

*Disadvantages of time-division multiplexing*

- (a) *Memory space distribution problem:* Every client creates a designated memory space, so clients who move from one unit to the next do not automatically receive a planned opening. A cell can be disconnected if, all of the time, allotments in the succeeding cell are now engaged. Similarly to that way, if all of the time allotments in the cell where a customer ultimately ended up are now occupied, the customer will not receive a phone line.
- (b) *Need for major modifications:* Increased data rates necessitated adjustment.

- (c) The effort required for synchronization: The synchronization burden is substantial. To maintain adequate signal transmission and reception, the transmitter and receiver components must be correctly and consistently synchronised.

### Exercise 3.5

1. Describe the types of multiplexing techniques.
2. Give an example of multiplexing in a real, non-computer-related system. Is the multiplexing centralised or distributed? Why? Give reasons.
3. Give the applications of a multiplexer.
4. What kind of network traffic is suited to synchronous time-division multiplexing?



### Activity 3.5: Identification of multiplexer and demultiplexer devices.

**Resources:** internet or library resources,

**Procedures:**

- (a) Visit internet or library
- (b) Search for multiplexers and demultiplex devices
- (c) From your search, identify any four devices on each
- (d) Describe the roles each identified devices and its communication mechanism

### Chapter summary

1. Data refers to the raw facts that are collected, such as text, image and symbols, while information refers to the processed data that are meaningful to the person who receives it, which enables us to take a decision. Data communications are the exchange of data between two devices via a transmission medium such as a wire cable and wireless (bound and unbound medium). A data communications system must transmit data to the right device accurately and promptly. The five components that make up a data communications system are: message, sender, receiver, medium, and protocol. Text, numbers, images, audio, and video are different forms of information.
2. Data communication systems can perform different tasks such as signal generation, interface, data synchronization, error detection and correction, flow control, addressing and routing.
3. The process for sending data between two devices connected over a network is referred to as transmission or communication modes. These modes are classified based on the number of wire connections and number of bits sent simultaneously, such as parallel and serial transmission; based on synchronization and time control, such as asynchronous, synchronous and isochronous; based on the direction of signal flow or direction of exchange of the information such as simplex, half-duplex and full-duplex.

4. A computer network is a collection of communication devices linked together using transmission media for data communication and resource sharing. Computer network elements include network devices, software, end-user devices and transmission media. Computer networks are basically classified based on geographical coverage, which are: PAN, LAN, MAN and WAN; network type is based on ownership/access restriction, based on topology, and based on architecture. Topology refers to the physical or logical arrangement of a network. Network devices may be arranged in a mesh, star, bus, or ring topology.
5. Computer network architecture refers to the design and implementation of a computer network. It is the physical and logical organization and arrangement of various network devices (i.e clients such as PCs, desktops, laptops and mobiles) to meet the needs of the end user/ customer. There are two main types of network architecture: peer-to-peer and client/server architectures. Client/server architecture can either be centralized or distributed.
6. There are two types of network models: the OSI model and the TCP/ IP model. OSI model consists of seven layers while TCP/IP consists of four. Each layer performs its specific tasks. Data encapsulation is adding additional information when data travels in OSI or TCP/IP model. Data de-encapsulation is the process of detaching information from the sent data as it travels from the physical layer to the application layer in the OSI model. TCP is a connection-oriented protocol while UDP is a connectionless protocol. An IP address is the logical numeric address that is assigned to every node within a network.
7. IP is of two types: IPv4, which can support a maximum of 4.3 billion addresses, and IPv6, which supports up to 340 undecillion. IP addresses are classified as class A, B, C, D and E.
8. Subnetting means dividing or separating a single network into multiple networks that can reduce the loading from one network. The subnet mask contains two fields, the network field and the host field.
9. Classless Inter Domain Routing (CIDR) is a method for assigning IP addresses without using the standard IP address classes like Class A, Class B or Class C.
10. There are two transmission media types: guided/wired and unguided/ wireless. Wired transmission media are subcategorized as coaxial cable, twisted pair cable and fiber optics cable, whereas unguided are categorized as infrared transmission, radio transmission and microwave transmission. The wireless network is the type of network whereby the type of communication medium is air.
11. Wi-Fi (Wireless Fidelity) is the industry-standard wireless local area network (WLAN) technology for linking computers and other electronic devices to one another

- and the Internet. Wireless network based on geographical area is classified as: Wireless Personal Area Network (WPAN); wireless local area network (WLAN); wireless ad hoc network (MANET); Wireless MAN, Wireless WAN, and cellular network. The main elements of wireless networks are wireless access point (WAP), wireless repeater, wireless router, wireless network card, wireless controller etc.
12. Some factors must be considered when choosing a wired or wireless network: network coverage requirements, network mobility, security, compatibility issues and the number and types of users.
13. Multiplexing is the process of combining multiple signals into one signal over a shared medium. There are mainly two types of multiplexing techniques, namely analog and digital. They are further divided into Frequency Division Multiplexing (FDM), Wavelength Division Multiplexing (WDM), and Time Division Multiplexing (TDM). Multiplexers are used in communication systems, computer memory and telephone network and transmission from the computer using satellite multiplexers. Demultiplexer is used on the receiving side to generate multiple signals from the received signal.

## PROJECTS

### Project 1: Ethernet cable construction and testing

#### Things to note

The making of the cable will depend on the purpose of the intended cable, whether for the same device connection, different device connection or configuration.

**Straight-through cable:** Is used to connect a different device such as a computer to switch, switch to router and switch to hub.

**Crossover cable:** Is used to connect the same device, such as computer to computer, switch to switch, router to router etc.

**Rollover:** Rollover cables, sometimes called Yost cables, are most commonly used to connect to a device's console port to make programming changes to the device.

The two standards for wiring Ethernet cables are T568A and T568B. The T568B is the most common and is the one you will be using for your Ethernet cable. Table 3.62 shows the proper orientation of the coloured wires to the pins.

#### Requirements

- Unshielded twisted pair (UTP) cable (length of the cable will depend on your application area)
- Modular connector (8P8C plug, also known as RJ45)

- (iii) Crimping tool
- (iv) Cable tester

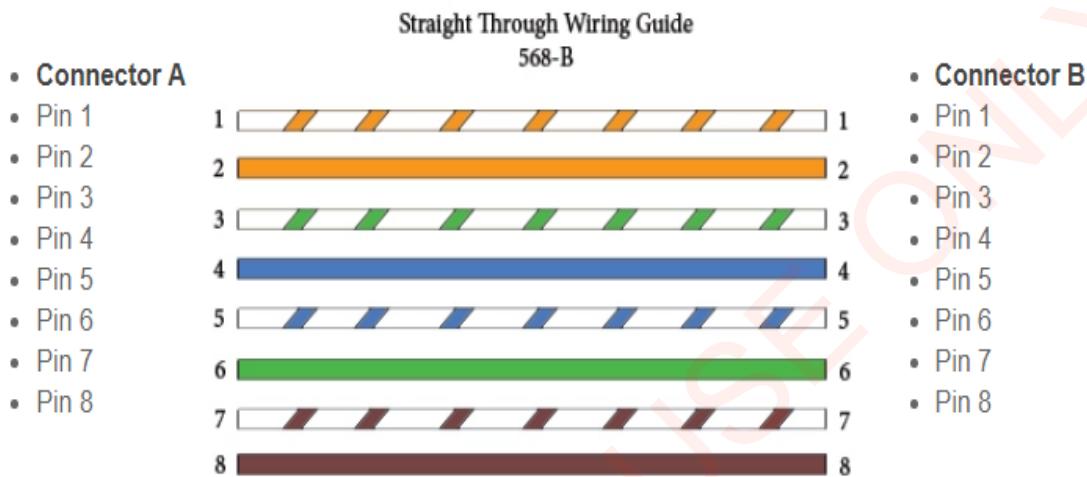
### **Procedure for making straight-through, crossover and rollover cable**

#### **Procedures for making straight-through cable**

Step 1: Strip the cable jacket about 1.5 inches down from the end

Step 2: Spread the four pairs of twisted wire apart

Step 3: Untwist the wire pairs and neatly align them in the T568B orientation (Figure 3.58). Be sure not to untwist them any farther down the cable than where the jacket begins.



**Figure 3.58: Straight through wiring guide**

Step 4: Cut the wires as straight as possible, about 0.5 inches above the end of the jacket.

Step 5: Carefully insert the wires into the modular connector, ensuring that each wire passes through the appropriate guides inside the connector.

Step 6: Push the connector inside the crimping tool and squeeze the crimper down.

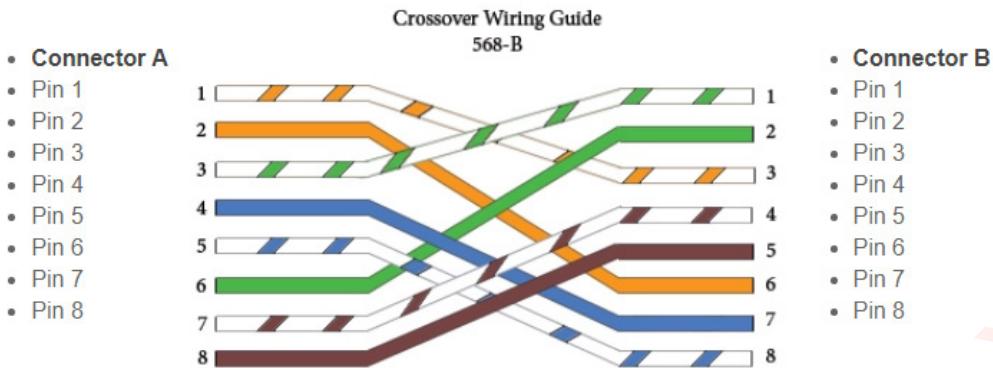
Step 7: Repeat steps 1-6 for the other end of the cable.

Step 8: To make sure you've successfully terminated each end of the cable, use a cable tester to test each pin.

#### **Procedures for making a crossover cable**

Follow the same procedure as for the straight-through cable above, except step 3 pin arrangement will be as on the Figure 3.59.

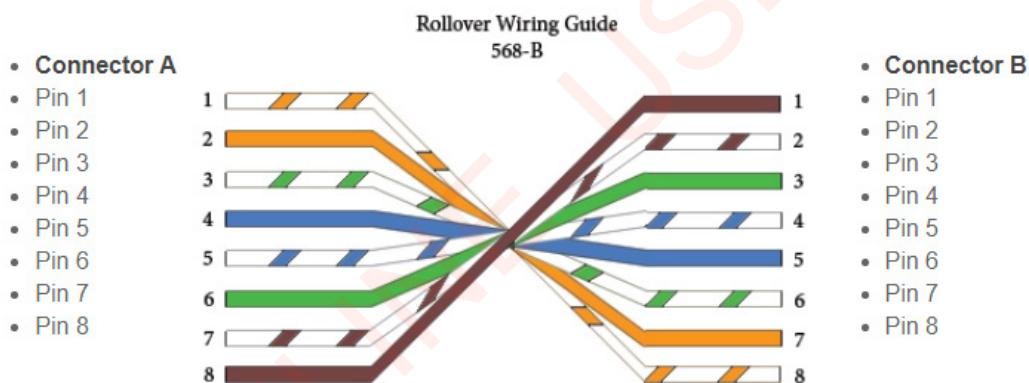
FOR ONLINE USE ONLY  
DO NOT DUPLICATE



**Figure 3.59:** Crossover wiring pins arrangement

### **Procedures for making a crossover cable**

Following the same procedure as for the straight-through cable above, except for step 3, the pin arrangement will be as shown in Figure 3.60.



**Figure 3.60:** Crossover cable

**Project 2:** Configure switch and router for internet connectivity

#### **A. Switch**

Basic switch configurations:

- Device name,
- VLANs
- Secure access control,
- Port configuration

#### **B. Router**

Basic router configurations

- Device name
  - Secure access control
  - Port security
- Routing configurations:
- Static and dynamic
  - Test for connectivity

**Revision exercise**

1. Describe the components of data communication systems.
2. Write short notes on the importance of data communication in education.
3. Contrast between:
  - (a) Digital and analog data
  - (b) Parallel and serial transmission
4. Describe the mechanism for character by character data transmission method.
5. Give the reason why asynchronous transmission takes more time than data transmission.
6. Account for the transmission method for transmitting data from the keyboard to the computer.
7. Describe the transmission method of the zoom online meeting
8. The contrast between parabolic and horn antenna mechanism
9. Describe the relative merits of client/server and peer-to-peer application models.
10. Classify each of the technology or systems as client/server, peer-to-peer or hybrid, and explain why the designers may have decided to implement the system in that way:
  - (a) eBay
  - (b) easybuy
  - (c) Skype
  - (d) BitTorrent
  - (e) SSH (Secure Shell)
  - (f) DNS
11. Based on question 10 for those which are peer-to-peer, explain how a new client joins the network and locates the data it is interested in.
12. Give three ways in which *synchronous* Time-Division Multiplexing is more complex than Synchronous Time-Division Multiplexing. Why is asynchronous TDM used on the Internet?
13. Contrast between TCP/IP and OSI model.
14. Describe the encapsulation and de-encapsulation process in OSI Model.
15. Contrast between classful and CIDR addressing scheme.
16. What is the network address if one of the host addresses is 220.220.20.89/24?
17. You have been allocated a class B network address of 135.1.0.0 and need to create 4 subnets each with around 200 hosts, what is the easiest mask to use to satisfy the criteria?
18. Describe the main configuration parameters which DHCP provides to nodes and the online purpose of each.
19. Compare and contrast DHCP with IPv6 autoconfiguration in terms of;
  - (a) how IP addresses are assigned to nodes;
  - (b) the configuration parameters provided to nodes;
  - (c) the interactions between nodes and the configuration server; and
  - (d) the design motivation.

20. Under what circumstances would you choose TCP, and when would you use UDP? In responding to this question, you might be interested in considering example cases such as:
- Fast transactions (where you want to send a short notification to another system with minimal latency)
  - Bulk data transfers
  - Timeliness (suppose data is being output from a sensor at one sample per second, and the receiver needs to have the most recent value of the sensor's reading rather than all values, i.e. the receiver should get the current value rather than an outdated value)
21. UDP provides very few features above those provided by IP, and is used by applications which do not require TCP's reliability and flow control (e.g. live video streaming). Why can such an application not just opt to use IP directly without a transport layer protocol?
22. Domain Name System (DNS) is an application layer protocol used to look up the corresponding IP addresses of host names such as `www.cl.cam.ac.uk` which originally used UDP only. Suggest why DNS might have been designed based on UDP rather than TCP.
23. Recently, the internet has started to deploy DNSSEC (DNS Security extensions), which involves transmitting cryptographic signatures and DNS data. These signatures are sometimes multiple kilobytes in size. Why might this motivate the use of TCP rather than UDP for DNS?
24. Discuss bounded and unbounded transmission media.
25. What are the factors to be considered in choosing transmission media?
26. For each of the following four networks, discuss the consequences if the connection cable fails.
- Five devices arranged in a mesh topology
  - Five devices arranged in a star topology
  - Five devices arranged in a bus topology
  - Five devices arranged in a ring topology

## Visual programming

### Introduction

Computer programming, which is the process of preparing computer programs, is at the heart of today's digital services. This is because almost every modern-day technology is driven by software. There are many tools and techniques for writing computer programs. In this chapter you will learn how to write computer programs visually (visual programming) to solve various problems using the Visual Basic .NET programming language. The competence generated will help you to write window-based programs.



#### Think about the following:

1. Accessing the command line (text) based program and GUI based program.
2. Developing a program using text based and Visual i.e. drag and drop techniques. Which one do you find ease to use or user friendly when you are required to creating GUI components, working with databases, and web programming?

### Concept of visual programming

Computer programs can be written in one of two ways: using the traditional text-based approach or using the visual programming approach. In text-based programming a programmer writes a program entirely in a text editor. This applies to both console as well as Graphical User Interface (GUI)-based programs. In contrast, visual

programming enables the programmer to write a program using both text and graphical components. For example, unlike in text-based programming where a programmer has to write code to create graphical components such windows and buttons, in visual programming these components are available for the programmer to use, usually by dragging and dropping. All the programmer has to do is to customise them and add business logic to a program.

Writing any GUI program of moderate complexity using the traditional text-based approach is difficult, as it requires a programmer to write a lot of code just to create graphical components, let alone writing code for solving the problem at hand. Thus, the major and obvious advantage of using visual programming techniques is to increase programmer productivity, that is, the programmer focuses on writing business logic code to solve the problem at hand instead

of worrying about creating graphical components. As creating graphical components requires a lot of code, visual programming offers a significant productivity boost to programmers.

### Examples of visual programming languages

Not all programming languages are visual in nature, but visual programming capabilities can be added to any text-based programming language. For example, the Basic programming language is text-based; when visual programming capabilities were added to the language it became to be known as Visual Basic. Similar capabilities are available in languages such as C++, Java, and Python. Visual programming capabilities are usually added to a programming language by means of software libraries together with integrated development environments (IDEs), for example the .NET (Visual Basic), QT (C++), Swing (Java), and PyQt (Python) libraries used in different IDEs to add visual programming capabilities in the respective languages. This chapter is based on Visual Basic .NET (VB. NET), a visual programming language that is based on the .NET framework.

### Importance of visual programming

Visual programming has the following importance:

- Visual programming is most useful when the consideration is in speed, automation, and ease of programming.
- The user interface are designed very quickly, using a mouse. The components are placed on the main

interface like forms and can be resized and moved more quickly.

- The developer may quickly progress in writing code without learning the syntax of the entire programming language. This results to graphic representation of the entire design process.
- Visual programming enables developers to test several new ideas at a rapid speed, and are comfortable to its graphical presentation flow, ease of use, and traceability.
- Visual programming offers the users the ability of programming without the necessity to learn code.
- With visual programming, users links the outputs of one segment to inputs of another, creating a program that moves from segment to segment by connectors more easily.
- Visual programming languages are easier to learn and use.

### Visual Basic .NET and the .NET Framework

This section discusses the VB. NET programming language and the .NET framework on which it relies for most of its functionality.

#### Visual Basic .NET

#### The concept of Visual Basic .NET

Visual Basic .NET is a modern object oriented programming language developed by Microsoft. The language is based on the .NET framework and was introduced in 2002 as a successor to the Visual Basic 6 programming language. In addition to object-oriented

capabilities, the language supports event-driven programming, that is, the ability of a program to respond to user actions such as mouse clicks. Although the language is a successor to Visual Basic 6, it is not compatible with the latter meaning that programs written in VB.NET will not compile in Visual Basic 6. Being built on top of the .NET framework, the language has full access to all the libraries and facilities offered by the framework including web programming frameworks, numerical libraries, and 3D graphics programming libraries.

### ***Features of visual basic .NET***

VB.NET has many attractive features that make it appealing to programmers. Some of its prominent features include:

- (a) *Easy to learn and use*: One can learn and be able to use the language to do something useful in a relatively short period.
- (b) *Automatic garbage collection*: The ability to automatically reclaim memory that is no longer in active use by a program, for example, when a data structure is no longer needed the memory it occupies is reclaimed and used for other purposes in the program.
- (c) *A rich standard library*: A collection of classes and frameworks ready for use by programmers in solving various programming problems. For example, components like buttons, check boxes and the like used in GUI programming.
- (d) *Properties and events*: Characteristics of components such as color and signals that notify components on the occurrence

of user activities such as mouse clicks. This enables a programmer to write code that is executed when the event occurs.

- (e) *Conditional compilation*: Selective compilation of blocks of code based on certain conditions. Conditional compilation is used when different versions of a program with different features need to be produced from the same source file; for example, when the same program is needed on two different platforms.
- (f) *Simple multithreading*: The ability for a program to have multiple parallel execution paths. This technique is often used to speedup programs by making use of multiple processors on a computer.

### ***Advantages of VB.NET***

Visual Basic .NET has the following advantages:

- (a) The .NET Framework is a software framework with a large library collection, which helps develop applications that are more robust.
- (b) Error handling is possible in the Visual Basic .NET Framework. It allows appropriate action to be taken where it encounters an error.
- (c) You can develop a simple program which works faster with a large desktop and web application using Visual Studio IDE.
- (d) It is an object-oriented programming language having objects and classes.
- (e) It employs drop and drag elements to create web forms in .NET applications.

- (f) Code will be automatically formatted in a VB.NET
- (g) Visual Basic .NET allows connecting one application to another created in the same language to run on the .NET framework.
- (h) You can connect your applications to other applications created in languages that run on the .NET framework.

### ***Disadvantages of VB.NET***

The following are some disadvantages of VB.NET:

- (a) It contains a large collection of libraries which holds a vast space in the system that takes more computing time during compilation.
- (b) It reduces jobs in the programming field as a VB.NET programmer because it is easy to learn resulting to more skilled programmers in the market.
- (c) The computer needs a JIT compiler to interpret the Intermediate Language source program, which requires an additional CPU cycle that degrades the performance of an application.
- (d) It requires a lot of programming, and it is not easy to manage every address by a pointer, resulting in additional coding, which takes more CPU cycles, and increases processing time. Hence, slows the VB.NET application.

### ***The .NET framework***

#### **Introduction to .NET framework**

.NET is a software framework that

simplifies the process of developing software. Software frameworks implement and make available to programmers commonly used programming functionality such as creating GUI components, working with databases, and web programming. The .NET framework provides a platform for developing desktop, mobile and web applications using different programming languages such as VB, C# (reads C sharp), and F# (reads F sharp).

To use the .NET framework for developing software you will need to install the .NET Developer Pack on your computer. You will also need to install an IDE to use in your development work. Microsoft provides an excellent IDE, Visual Studio (VS) 2022, for this purpose. VS 2022 comes bundled with the .NET framework, so if you install it you will not need to install the .NET framework separately.

#### **Components of .NET framework**

The .NET Framework is composed of the following components:

- (a) .Net Framework Class Library: It consists of a huge library of reusable types; the interfaces, classes, structures, and enumerated values, together called types.
- (b) Common Language Runtime or CLR: It is responsible for memory management, debugging, code execution, code safety, exception handling, thread execution, security checking, verification, and compilation. It converts the program into native code. It acts as an interface between the framework and operating system.

- (c) Windows Forms: This contains the graphical representation of any window displayed in the application.
- (d) ASP.NET and ASP.NET AJAX: It is used to develop websites, web applications, and web services. It provides a strange integration of HTML, CSS, and JavaScript. It contains AJAX, an extension for developing and implementing AJAX functionality. ASP.NET AJAX contains the components that allow the developer to update data on a website without reload the page completely.
- (e) ADO.NET: It is the technology used for working with data and databases. It provides access to data sources like SQL server, OLE DB, XML etc. The ADO.NET allows connection to data sources for retrieving, manipulating, and updating data.
- (f) Common Language Specification: It contains the specifications for the .Net supported languages and implementation of language integration.
- (g) Common Type System: It provides guidelines for declaring, using, and managing types at runtime, and cross-language communication.
- (h) Metadata and Assemblies: Metadata is the binary information describing the program, which is either stored in a portable executable file (PE) or in the memory. Assembly is a logical unit consisting of the assembly manifest, type metadata, IL code, and a set of resources like image files.
- (i) Windows Presentation Foundation: It provides a separation between the user interface and the business logic. It helps in developing visually stunning interfaces using media, documents, animations, 2-D and 3-D graphics, and others.
- (j) Windows CardSpace: It provides safety for accessing resources and sharing personal information on the internet.
- (k) Windows Workflow Foundation (WF): It used in building workflow-based applications in Windows. It consists of activities, workflow designer, workflow runtime, and a rules engine.
- (l) Windows Communication Foundation (WCF): It is the technology for developing and executing connected systems. It is a framework for building service-oriented applications. It can assist in sending data as asynchronous messages from one service endpoint to the next.

### Features of .NET framework

.NET Framework contain several features, which make it better, and more ease to modern programming activities. The following list describes .NET Framework main features:

- (a) Interoperability: Normally, interaction between new and old applications is required, so, .NET Framework provides a way to access features that are implemented in programs that run outside the .NET environment.
- (b) Support Multi languages: The .NET has a Common Type System

- (CTS) that defines all possible data types and programming constructs supported and provides the means how they interact with each other. This allows .NET to support multiple application development programming languages, including C#, or Visual Basic.
- (c) Portability: Applications written using .NET language are portable. That means that when the source code of a program written in a Common Language Runtime (CLR) compliant language compiles, it generates machine-independent, intermediate code. It is also possible to mix and match code written in the languages supported by the framework.
  - (d) Base Class Library: The Base Class Library (BCL) is a type library available for all languages that use .NET. The BCL provides classes that encapsulate many common functions, including reading and writing files, database interaction, image design, XML and JSON manipulation, etc.
  - (e) Common Language Runtime (CLR): All the programming Languages in .NET are compiled into an intermediate language known as Common Intermediate Language (CIL). This intermediate language is not interpreted but compiled to native code in a way known as Just In Time compilation (JIT). The combination of these is called Common Language Infrastructure (CLI).
  - (f) Automatic Resource Management: .NET CLR automatically manages memory, database connections, network, etc. It invokes built-in functions to allocate and deallocate the memory used by objects during runtime. This removes the developer's burden of managing memory.
  - (g) Deployment made easy: When it is time for application deployment, depending on the desired deployment environment (mobile, desktop, cloud, etc.) the developer has abundant options to package the application.
  - (h) Security: .NET allows code to run in different levels of security without separate security mechanism.
  - (i) Easy and rich profiling and debugging support: Integrated Development Environments (IDEs) like Visual Studio or Visual Studio Code provide a place to develop and debug .NET applications. The framework provides rich debugging and profiling information, which is highly integrated within these IDEs. This helps us to check the application performance and track down runtime exceptions.

### Installing Visual Studio (VS) 2022

Before you can start programming in VB.NET you need to install the VS IDE. It is recommended that you install the latest version. In this book VS 2022 will be used. There are about five steps in installing VS 2022. While these steps are mandatory, also some are optional. To install VS 2022, follow the following steps.

1. Check your computer for readiness: Before installing VS on a computer,

## FOR ONLINE USE ONLY

- you should ensure the computer meets the following requirements:
- Windows 10 or higher on an x64 processor with at least 1.8 GHz and a minimum of 4 cores. In addition, at least 4 GB of RAM are required and a minimum of 850 Mb of hard disk space (an SSD hard disk is recommended). Finally, the computer should have a video card with a resolution of at least 1366 x 768.
2. Update Windows: Update the Windows Operating system to ensure it includes the latest security updates choosing continue (see Figure 4.1.)
  3. Download the VS bootstrapper file: The bootstrapper is a small file that when run downloads the VS installer. Go to <https://visualstudio.microsoft.com/downloads> to download the file. Choose the community edition of VS. The file will have a name similar to *vs\_community.exe*.
  4. Run the bootstrapper file: To download the VS installer, run the bootstrapper file. Agree to the terms stated on the shown dialog box by

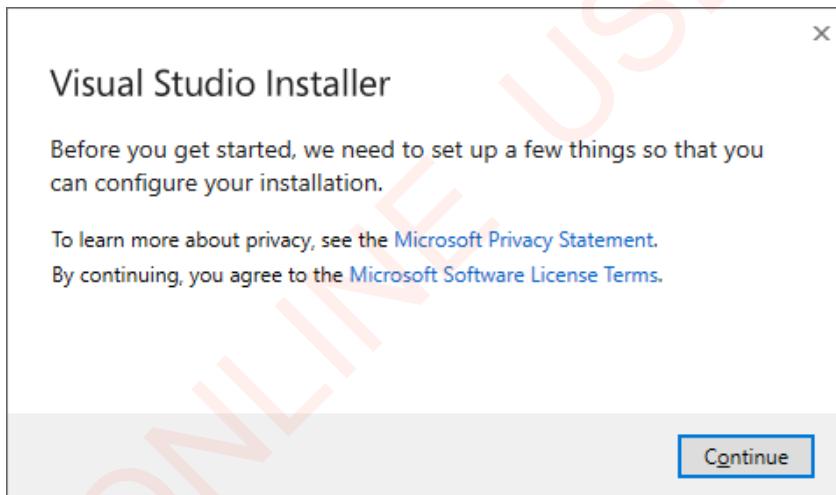
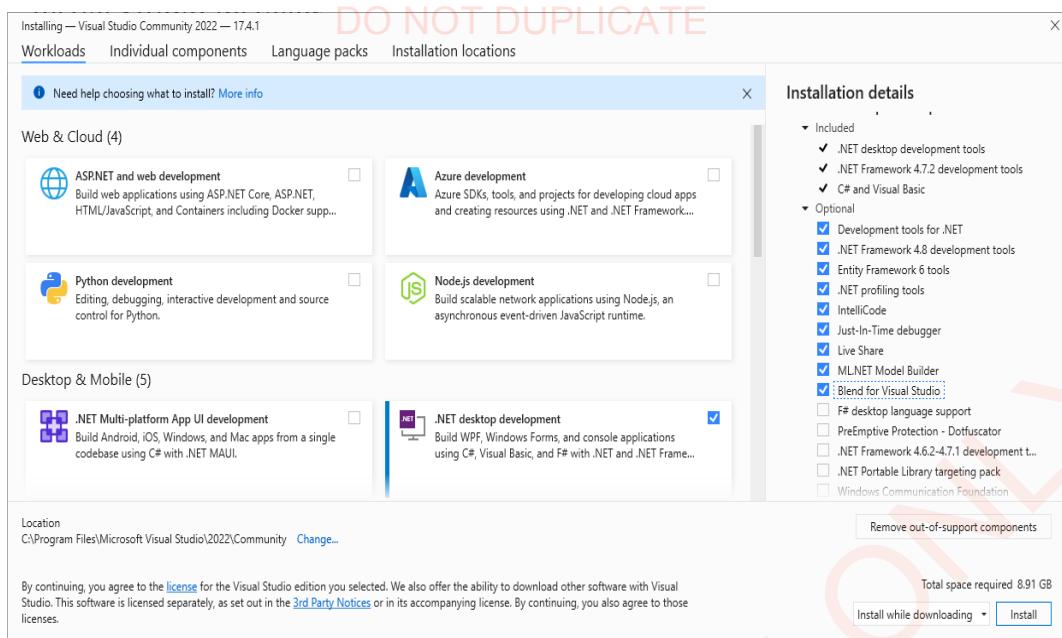


Figure 4.1: Running the VS bootstrapper file

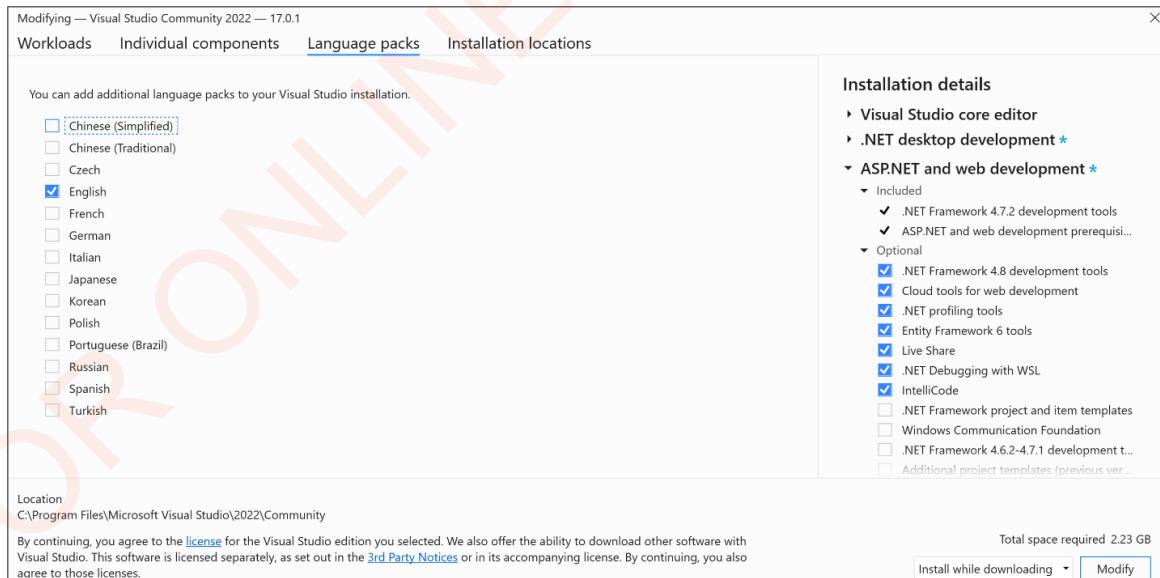
5. Choose features (workloads) to install: When the installation is complete select the features you want to install; In this case choose the .NET desktop development option (see Figure 4.2.)
6. Choose install to start the installation process.

FOR ONLINE USE ONLY  
DO NOT DUPLICATE



**Figure 4.2:** Selecting VS features to install

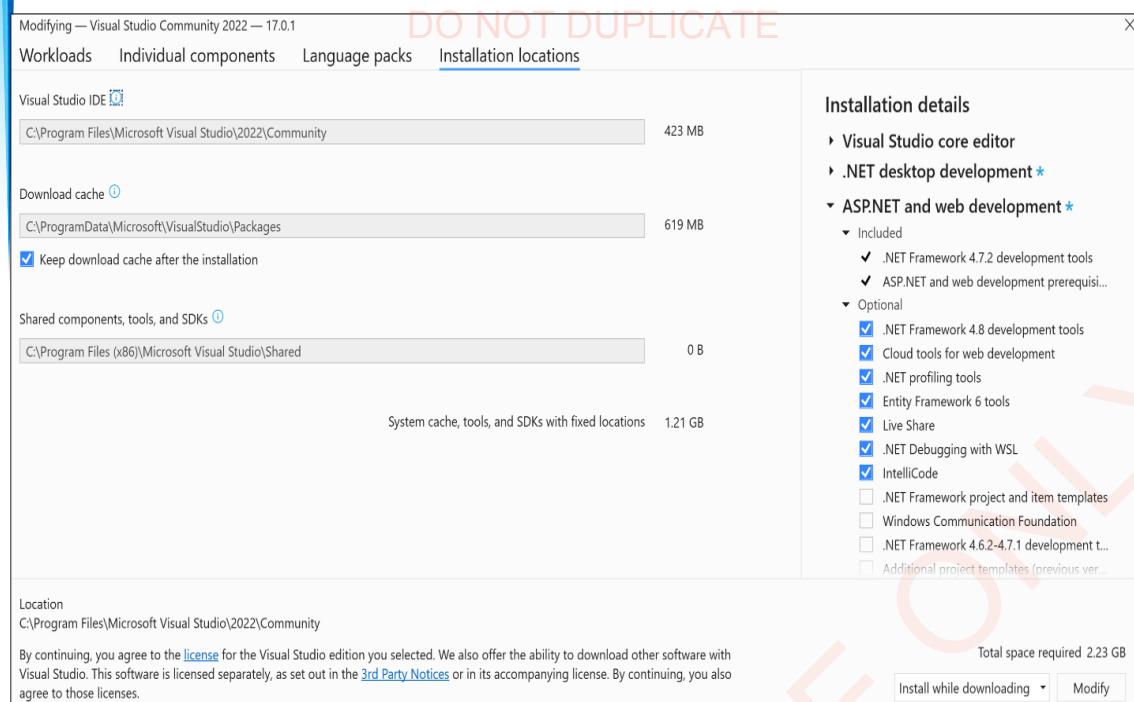
The steps described above are necessary for installing VS 2022. Optionally, you can change the installed language or install additional language packs alongside the default language. The installer chooses the default language based on the operating system language (Figure 4.3 (a)). Similarly, you can choose a different installation location as shown in Figure 4.3 (b).



**Figure 4.3: (a)** Installing additional language packs/changing installation language.

FOR ONLINE USE ONLY

DO NOT DUPLICATE



**Figure 4.3 (b):** Choosing a different installation location

After VS is installed, you can launch it and explore its features. To do this, perform Activity 4.1.



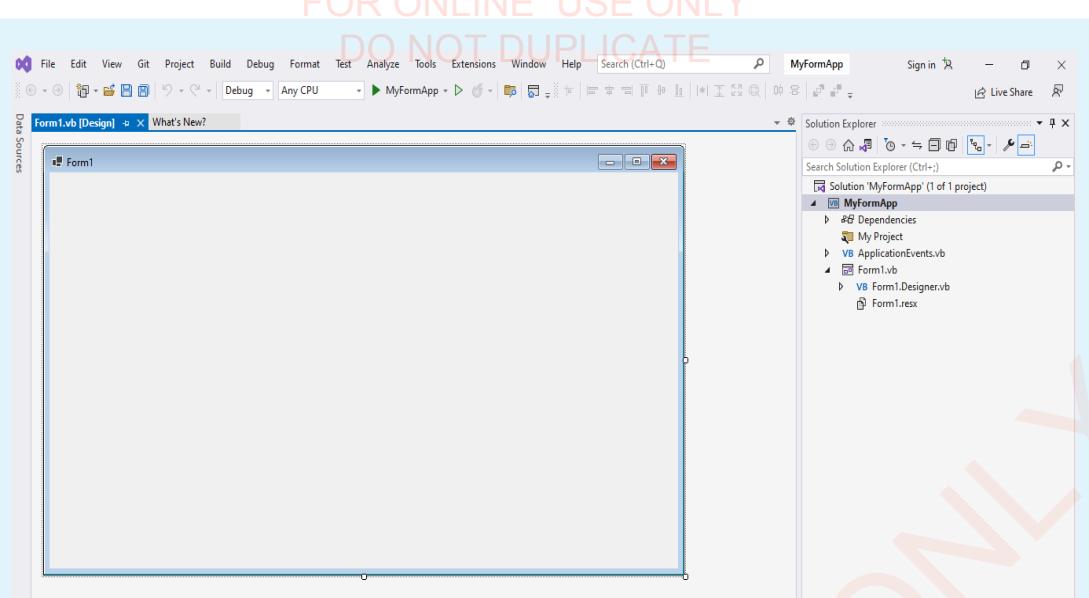
### Activity 4.1: Exploring the VS 2022 IDE

**Resources:** A computer with VS 2022 IDE

**Procedure:**

1. Launch VS 2022 and carefully study the features of the start window shown. Try to understand the different options for creating and opening projects.
2. Choose **Create a New Project** to create a new Windows Forms application. Choose VB for language and Windows Forms application (.NET) for application type. Fill in the basic details and options about the application such as project name.
3. Create the project. You will now have a new Windows Forms project to work with. In the next section you will learn how to work with different controls. The final IDE window after completing these steps should look similar to the one shown in Figure 4.4.

FOR ONLINE USE ONLY  
DO NOT DUPLICATE



**Figure 4.4:** A newly created Windows Forms project

### Exercise 4.1

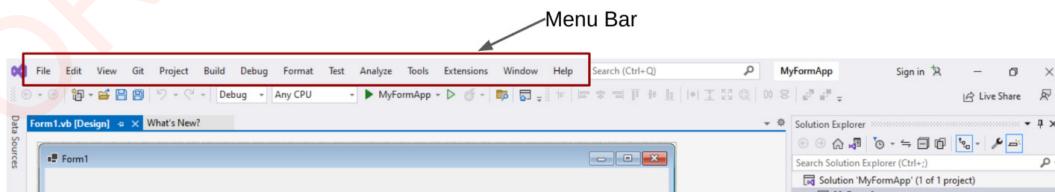
1. What do you think are the drawbacks of visual programming?
2. Contrast between VB.NET and visual programming.
3. Is visual programming suited for building every kind of software? Explain.
4. List the two elements of a Visual Basic Application.
5. What is the purpose of a GUI and what elements does a user see in a GUI?

## Components of the Visual Studio 2022 IDE

This section explains the different components and tools of the Visual Studio IDE that will allow you to create Visual Basic .NET applications.

### Menu Bar

Menu Bar , shown in Figure 4.5, is the Visual Basic command to create, complete, and execute your application.



**Figure 4.5:** The Menu bar in the Visual Basic IDE

## Visual Basic .NET Controls DO NOT DELETE

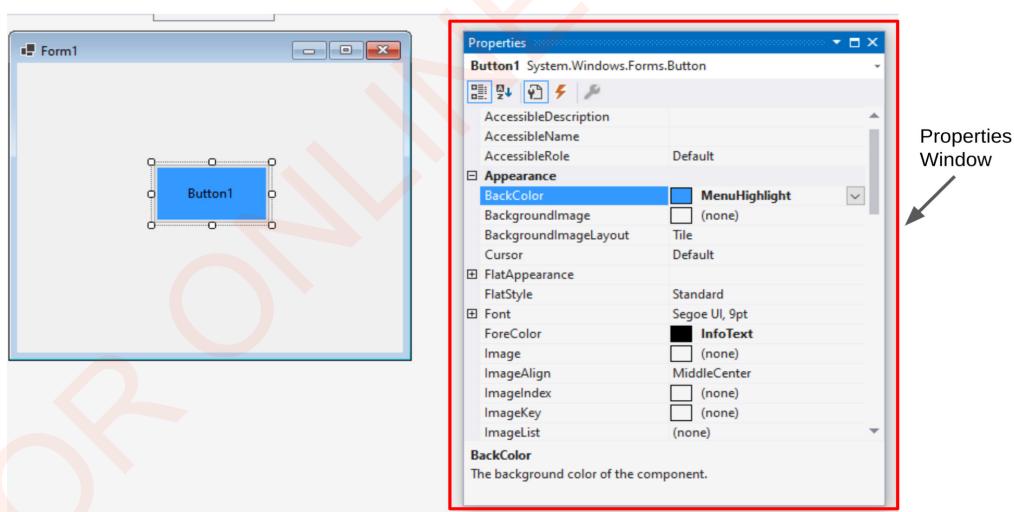
Controls are objects placed on a Visual Basic form. A form is also a control object that acts as a container that hosts other controls of the application. Each control has its own set of properties, methods, and events that can be configured by the programmer depending on their needs. The toolbox is a window in Visual Studio that provides all controls. To launch the toolbox, click View on the Menu Bar and then select Toolbox. The Toolbox window will appear, with several tabs that contain different controls. Pay special attention to the Common Controls tab. To add controls to a form, simply double-click on it or drag and drop it on the form.

### Visual Basic .NET Control elements

Visual Basic has three control elements, which are control properties, control methods and control events.

Properties describe the control in terms of appearance, layout and behaviour. For instance, each control can have a name, size, label, font size and font type. Control properties can be set using the properties window or using visual basic code. To edit the properties of the control once it is on the form, right-click on the specific control and select *Properties*. It will open up a properties window that allows you to set several properties of the control.

For example, Figure 4.6 shows the properties window when a Button control is selected. Using this Properties window one can set the appearance, layout, behaviour, accessibility of the button and more. For example, you can change the text's font, background colour, point size, and alignment of the Button control.



**Figure 4.6:** Properties window in Visual Studio

## Control Methods

A method is a procedure that performs a certain task. Controls in visual basic have methods which perform certain tasks by accessing information within the control class. Methods direct the object to do something. All controls have built-in methods that can just be called. However, if none of the methods performs the task needed, new methods can be created.

For example, the control *Textbox* has a built-in method called *Text()* whose function is to get or set the input text of the *Textbox*.

## Control Events

Control events notify the control that an event has occurred. Examples of events are click, double click, close, resize and load. When a certain event occurs, the specific method that handles that event is called. For example, when a user clicks on a Button control, the event method *Button\_Click()* is called. All code that needs to be executed when a button is clicked should therefore be written in this event handling method (*Button\_Click()*) as shown in Figure 4.7.

```
1 reference
Public Class Form1
    0 references
    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
        MsgBox("You have clicked a button")
    End Sub
End Class
```

**Figure 4. 7:** A button click event that handles the click of a button.

In this example, from Figure 4.7, a button click event handles the click of a button when *Button1* is clicked; a message box appears with the text “You have clicked a button”.

## Visual Basic .Net Toolbox controls

### Common/Basic Controls

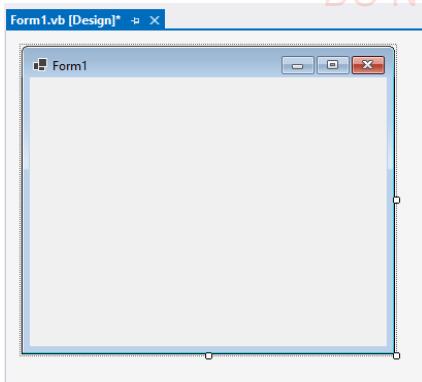
#### Forms

Visual Basic allows us to create form-based applications. Each window on the desktop that a Visual Basic application displays is a form. The words form and window both refer to the same thing. Therefore, a form is a container for all the controls that make up the graphical user interface of visual basic applications.

To create a form-based application in Visual Studio, click on File > New Project and then select *Windows Forms Application* project. Once the project is created, a new blank form will appear as in Figure 4.8.

The Controls and their functions are described in Table 4.1.

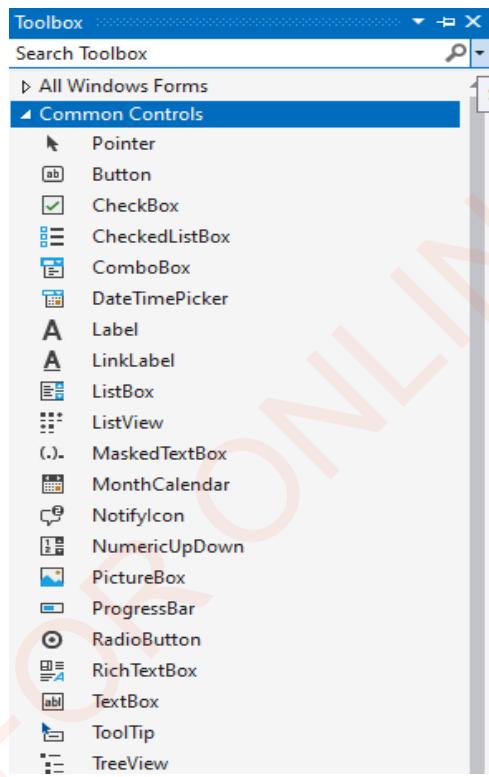
**Table 4.1:** Controls and their description



**Figure 4.8:** A form in VB .NET

### Elements of the VB .NET ToolBox

In this book, we will discuss the common controls that are frequently used. Figure 4.9 shows the common controls and a brief description of what each component does is given in Table 4.1.



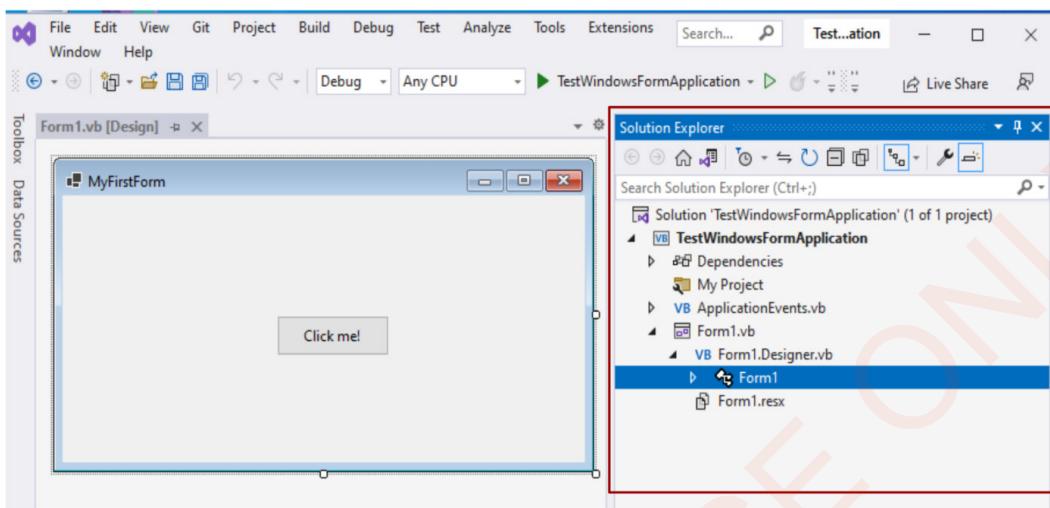
**Figure 4.9:** Toolbox showing common controls that can be added to forms

| Control          | Function                                                                                                                                                                   |
|------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Button           | A control that represents a button for a user to press.                                                                                                                    |
| Check Box        | A control that provides the user with a toggle choice (checked or unchecked).                                                                                              |
| Checked List Box | A control that provides a user with a list of options to select From. It allows the user to select one or more boxes.                                                      |
| Combo Box        | A control that displays a drop-down list of various items. It combines a text box where the user can add the item and a drop-down list where the user chooses added items. |
| Date Time Picker | A control that allows users to select a time and date on a form                                                                                                            |
| Label            | Allows adding text to the form                                                                                                                                             |
| Link Label       | This control allows you to add links (web-based like) to the form. You can link to a file or a folder webpage.                                                             |
| List Box         | A control that works to display a list of items on the application. Users can select any options from the list.                                                            |

| Control         | Function                                                                                                                                                                                                                               | Control       | Function                                                                                                                                                                                                                                                                   |
|-----------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| List View       | This control allows displaying a list of items with icons                                                                                                                                                                              | PictureBox    | A control that is used to display images and graphics. The image can be in any supported format and size such as, Bitmap (.bmp), JPEG (.jpg), icon (.ico), the cursor (.cur), run-length encoded (.rle), metafile (.wmf), enhanced metafiles (.emf), and GIF (.gif) files. |
| Masked Text Box | A Control that allows for both restricted data entry and formatted data output. This control provides visual indications of the type of data entered or displayed.                                                                     | Radio Button  | A control that limits the user to pick just one option. The programmer can set any of the buttons as default if needed.                                                                                                                                                    |
| Month Calendar  | A control that allows the selection of months and data from a calendar. It also allows the selection of a range of dates on the calendar.                                                                                              | Rich Text Box | A control that allows formatting text, e.g., making the text bold, italics, underlining or adding colour. It also allows the displaying of formatted text.                                                                                                                 |
| Notify Icon     | This control display icons for processes running in the background. The NotifyIcon Control is used to show the status of background applications.                                                                                      | Text Box      | A control that is used to accept textual input from the user. The user can add strings, numerical values and a combination of those.                                                                                                                                       |
| Numeric Up Down | This control consists of a text box with a number and a pair of arrows (upward and downwards). It allows the user to increase the numerical value by clicking the upward arrow and decrease the number by clicking the downward arrow. | Tool Tip      | This control enables the display of text when the user points at any control. It can be used with any control and is commonly used to provide users with more information or help tips.                                                                                    |
| TreeView        | A control that displays a hierarchy of nodes.                                                                                                                                                                                          |               |                                                                                                                                                                                                                                                                            |

## Solution Explorer

The solution explorer (Figure 4.10) is a special window in the Visual Basic IDE that allows you to see all your opened projects as well as, all files and elements that your projects contain. The solution explorer is similar to the project explorer or project window in previous versions of Visual Basic.

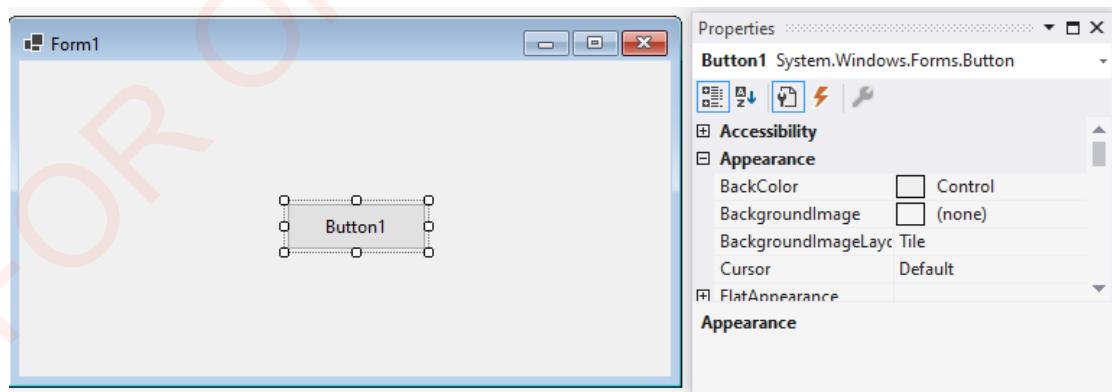


**Figure 4.10:** The solution explorer window in Visual basic

## Editing form controls

To add any of the controls described in Table 4.1, you can either drag and drop the control from the toolbox view to the form or double-click on the control, which will appear on the form. To edit the control from the form, right-click on the specific control and select **Properties**. This will open up a properties window that allows you to set several parameters of the control.

For example, Figure 4.11, shows the properties window when a button control is selected. Using this **Properties** window one can set the appearance, layout, behaviour, accessibility of the button and more. For example, you can change the text's font, background colour, point size, and alignment of the button control.

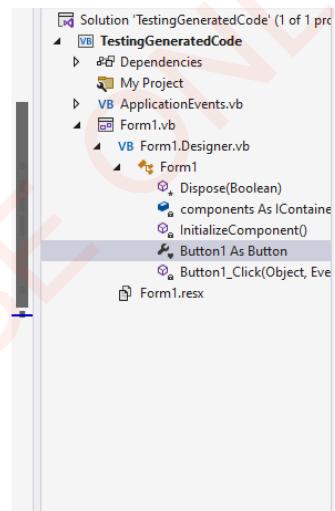


**Figure 4.11:** Properties window in Visual Studio

**Design View and Code View** **DO NOT DUPLICATE**

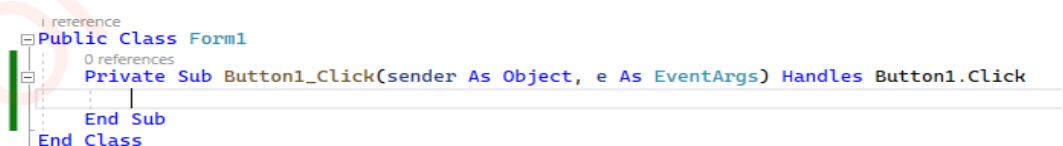
The advantage of visual programming is that the programmer adds the controls using a graphical user interface (by dragging and dropping) and the IDE generates the necessary code needed to create that control. For example, adding a Button control will generate the code needed to create and show the button. To see the code that has been generated, right click on the control that you want to edit and select *view code*. Use the solution explorer to navigate to the specific component (in this case Button1). This will open a *Designer.vb* view which shows all the code that the IDE has automatically generated as shown in Figure 4.12. Note that the code in the *Designer.vb* view is not meant to be edited, instead to edit visual controls use the properties view.

```
'NOTE: The following procedure is required by the Windows Form Designer
'It can be modified using the Windows Form Designer.
'Do not modify it using the code editor.
<System.Diagnostics.DebuggerStepThrough()
0 References
Private Sub InitializeComponent()
    Me.Button1 = New System.Windows.Forms.Button()
    Me.SuspendLayout()
    '
    'Button1
    '
    Me.Button1.Location = New System.Drawing.Point(180, 114)
    Me.Button1.Name = "Button1"
    Me.Button1.Size = New System.Drawing.Size(75, 23)
    Me.Button1.TabIndex = 0
    Me.Button1.Text = "Button1"
    Me.Button1.UseVisualStyleBackColor = True
    '
    'Form1
    '
    Me.AutoScaleDimensions = New System.Drawing.SizeF(7.0!, 15.0!)
    Me.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font
    Me.ClientSize = New System.Drawing.Size(800, 450)
    Me.Controls.Add(Me.Button1)
    Me.Name = "Form1"
    Me.Text = "Form1"
    Me.ResumeLayout(False)
End Sub
Me.ResumeLayout(False)
```



**Figure 4.12:** Designer view showing code generated for a button

While the Visual Studio IDE allows you to drag and drop controls to your form applications and edit these controls using the properties window, the code view allows you to add specific functionalities to these controls. For example, to add code that should be called when the Button is clicked, double click on the button in the design view as shown in Figure 4.12. This will open a *Formname.vb* code view which allows you to add code that controls events of the button, for example the *onClick* event as in code snippet in Figure 4.13.



**Figure 4.13:** Code view showing a Button click function

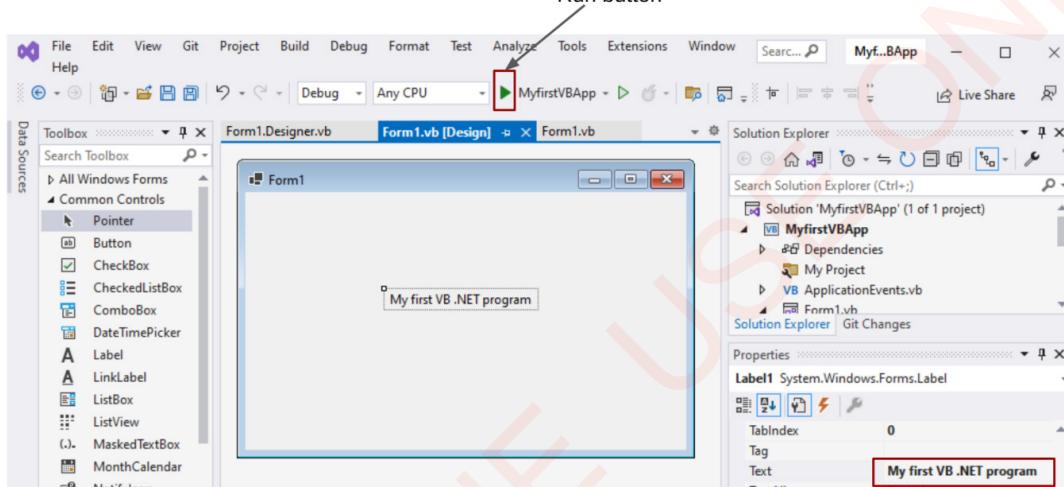
## FOR ONLINE USE ONLY

## Creating your first application in Visual Basic .NET

Now that we have explored the Visual Studio IDE and covered which controls are available, it is time to create a full working application using this knowledge. We start by creating a simple form that will display the text “Welcome to my Visual Basic .NET application”.

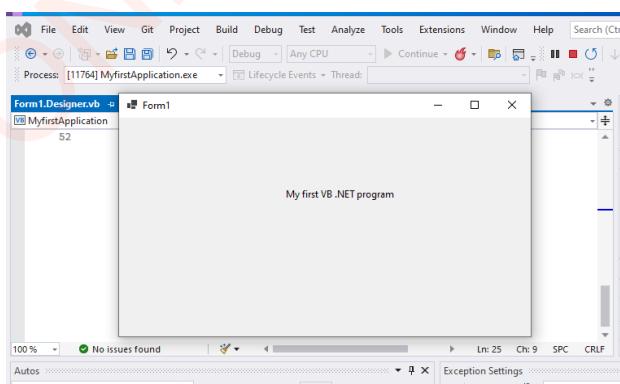
Open your Visual Basic IDE and create a new Windows Forms Application

project. A window containing a new form will appear. Open the toolbox (**View > Toolbox**), then drag and drop the control Label property to the new form. Open the properties window and under appearances, look for the named Text and type in “My first VB.NET program” (Figure 14). Click on the run button to run the application. The form should appear with the text “My first VB.NET program” as shown in Figure 4.15.



**Figure 4.14:** The design view in Visual Studio showing a form and a label control

The output of the program will be shown in Figure 4.15.



**Figure 4.15:** Output of the program when it is run

## Exploring features and basic controls of Visual Basic .NET

To explore how to use more features and controls of Visual Basic .NET to create applications, we will use an example of an application that will manage the registration process of Form 5 students to a school. The application will require the students to fill in a registration form with information such as first name, last name, gender, date of birth, combination and add their picture. The controls that will be used to create this registration form are explained hereafter.

### Labels

A label is used to display fixed text and numbers on the form. It provides instructions to the users and displays the output.

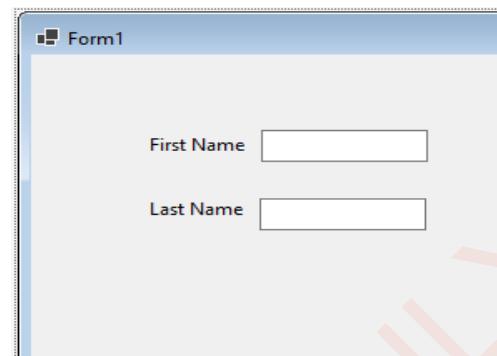
### TextBox

The TextBox is a box for entering and displaying text (characters or values) in a user interface. This tool is used frequently in most applications. In form applications, usually, a label shows the name of the text box.

### Example

Design a registration form that allows Form 5 students to enter their firstname and lastname. Use the toolbox to drag two text boxes and two labels as shown in Figure 4.16. Change the following properties using the property window, *Name*, which reflects how the control will be referred to in the code and *Text* which will appear on the screen when the program is run. You do not need to add any value to the *Text* property of the TextBox, this is usually left empty for the user to fill in. Figure 4.16 shows this example on the graphical

view. To capture what the user enters in the TextBox, use the command *textboxname.getText*.



**Figure 4.16:** Label (First Name and Last Name) and two empty TextBoxes

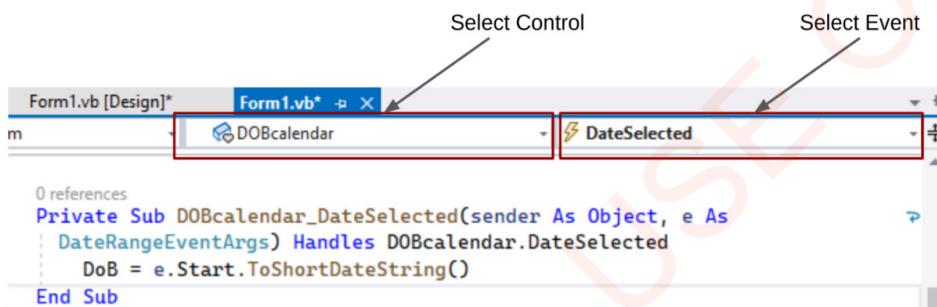
### MonthCalendar

This control allows the user to select year, month and date from a calendar view (Figure 4.17). In the registration form, we will use the MonthCalendar control to capture the date of birth of the student. Drag and drop the MonthCalendar control from the toolbox and use a label to indicate that the student should enter their birth date using the calendar. The MonthCalender control has many properties that can be configured, for example you can set the minimum and maximum date that can be selected. To capture the date selected by the student, right click on the calendar and click code view. Select the calendar name and then select DateSelected, this will generate a block of code where you can capture the date selected and use it in the program. In our application, a variable DoB of type String is defined, and the selected date of birth is stored on this variable as shown in Figure 4.18.



**Figure 4.17:** An example of a calendar as it appears on the form

The code snippet in Figure 4.18 captures the selected date.



**Figure 4.18:** Visual basic code used to capture the selected date

### Listing options

Visual basic .NET contains four controls that can be used to list options to the user on a form. These are RadioButton, ListBox, checkedListBox and ComboBox. We discuss when it is appropriate to use which one using our registration form example.

#### RadioButton

A radio button allows the user to select one out of many available options. It is used to display a list of mutually exclusive choices. In this example, the student needs to select a gender and they can either select Male or Female. Use the toolbox to drag two radio buttons and use the property window to

edit the Text property and set them to Male and Female respectively as seen in Figure 4.19.

The radio button allows you to capture a selected option and assign it to a variable for use in the program. For example, in this application, we have created a variable named Gender of type String which is set to male when the user selects the male radio button and set to female when the user selects a female radio button.



**Figure 4.19:** Radio buttons for choosing gender

## FOR ONLINE USE ONLY

The code snippet in Figure 4.20 captures the radio button selection.

```

1 reference
Public Class Form1
    Dim gender As String
    0 references
    Private Sub RadioButton1_CheckedChanged(sender As Object, e As
        EventArgs) Handles malebutton.CheckedChanged
        gender = "male"
    End Sub
    0 references
    Private Sub RadioButton2_CheckedChanged(sender As Object, e As
        EventArgs) Handles femalebutton.CheckedChanged
        gender = "female"
    End Sub

```

**Figure 4.20:** Visual basic code to capture the radio button selection

### List Box

This control allows listing of several options to the user and the user can select one or more. For example, if we want the student to select one or more combinations of subjects from a fixed set of choices, a ListBox is appropriate. Use the toolbox to drag and drop a list box in the form. Select the ListBox added and go to the properties window. In the properties window, go to the Items property and click on the button with three dots on it. A new window will appear allowing you to list the options that should be on the ListBox. In Figure 4.22, the list contains four choices which are PCM, PCB, PMC and MCSE. To allow selection of multiple items on the ListBox, set the SelectionMode property to MultiSimple.

To capture what has been selected on the list box, right click on the ListBox, and click view code. Select the list box and in the events box select SelectedValueChanged. This will add a block of code which allows you to capture the selectedItems on the list (see Figure 4.21).

```

Dim selectedcombinations As String = ""
0 references
Private Sub listboxCombinations_SelectedIndexChanged(sender As Object, e As EventArgs) Handles
    listboxCombinations.SelectedIndexChanged
    For Each selectedItem As Object In listboxCombinations.SelectedItems
        selectedcombinations &= selectedItem.ToString + " "
    Next
End Sub

```

**Figure 4.21:** Visual basic code to capture the selected items.

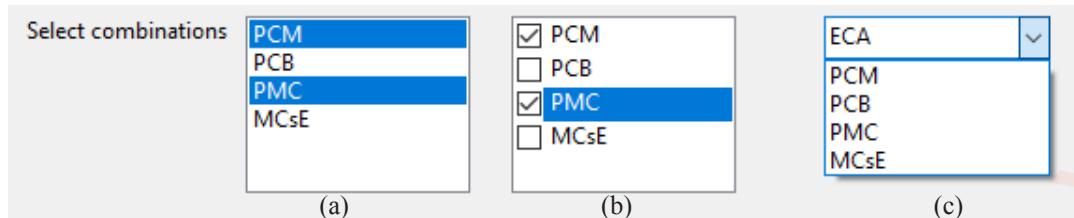
### CheckedListBox

A CheckedListBox is similar to a ListBox except that the CheckedListBox has a check box next to each list item and this check box can be checked with a tick. The middle box in Figure 4.22 shows an example of a CheckedListBox.

## FOR ONLINE USE ONLY DO NOT DUPLICATE

### ComboBox

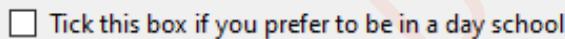
This control also allows selection of choices. However, the comboBox allows the user to enter new choices that are not provided in the list. In Figure 4.22, the comboBox shown allows the student to add a new combination that was not listed in the choices.



**Figure 4.22:** (a) The *ListBox*, (b) *CheckedListBox* and (c) *ComboBox*.

### Check Box

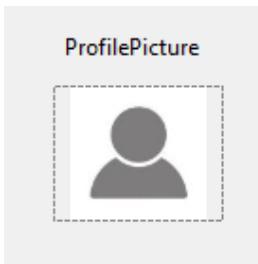
This control can be used to show one option that a user can select or deselect. The CheckBox has a property called checked which is set to *True* when the checkbox is selected and *False* when unselected. In Form 5 application example, a CheckBox is used to let a student choose whether they prefer to be in day school instead of boarding school as shown in Figure 4.23.



**Figure 4.23:** An example of a *CheckBox*

### Picture Box

This is a control that is used to add images into a form. In this example, we want the student to upload their picture and then display it on the form. To achieve this, a PictureBox control that will hold that image is needed. Use the toolbox to add a PictureBox into the form and a label to indicate that it will display a profile picture as shown in Figure 4.24. A default picture can be set by using the image property in the properties window which allows you to navigate and select a default image.



**Figure 4.24:** An example of a *PictureBox* with a default image.

## Button

A button acts as a switch which is used to run a specific block of code. The Button control is used to link and call a function in a program.

In the registration form example, the Button control is used to capture all information a student has filled in the form and display it on a message box.

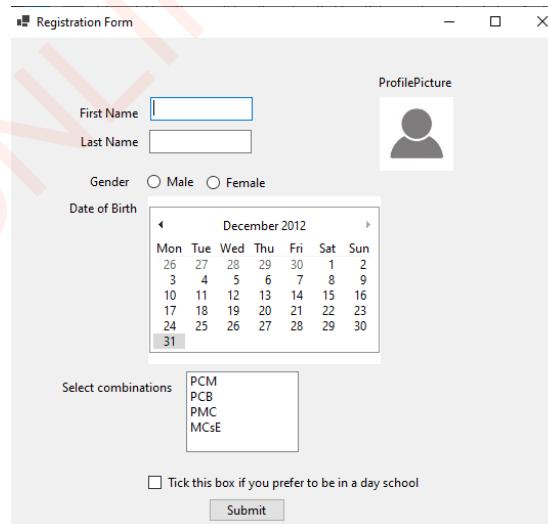
Drag and drop a Button control to the form. Change the Text property of the button from *Button1* to *Submit*. Also

change the Name property of the Button from *Button1* to *SubmitButton*. Right now the button exists but does not have any functionality. Right click on the button and click **View code**. The code editor will open with the cursor placed where the code for controlling the button needs to be. Add the code below to show a message box that captures the first name, last name, gender, date of birth, selected combination of the student and whether the student prefers day school or not as shown in Figure 4.25.

```
0 references
Private Sub Submit_Click(sender As Object, e As EventArgs) Handles SubmitButton.Click
    MessageBox.Show("First name: " + firstname.Text + " Last name: " + lastname.Text +
    " Gender: " + gender + " Date of Birth: " +
    " DoB + " Combinations selected: " + selectedcombinations +
    " Prefers Day school? : " + daySchoolCheckBox.Checked.ToString)
End Sub
End Class
```

**Figure 4.25:** Visual basic code to capture input from the form and display it on a message box.

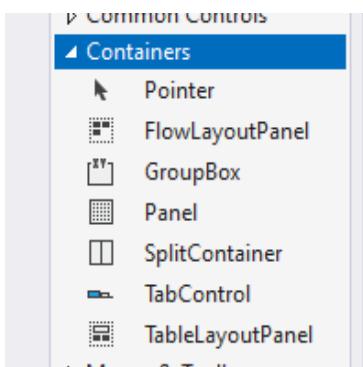
Figure 4.26 shows the complete Form 5 registration program application that has been developed using different features and controls available in VB. NET.



**Figure 4.26:** Student registration form developed using Visual Basic .NET

## Containers

Containers are controls that have the ability to host other controls. Forms are the base containers since they host all other controls. Apart from Forms, Visual Basic consists of other container controls. These container controls can be found in the Containers Tab of the ToolBox. The container is referred to as the parent and the controls it contains are referred to as Children. In this book, we will describe two common containers, The GroupBox and TabControl. Figure 4.27 shows the container control



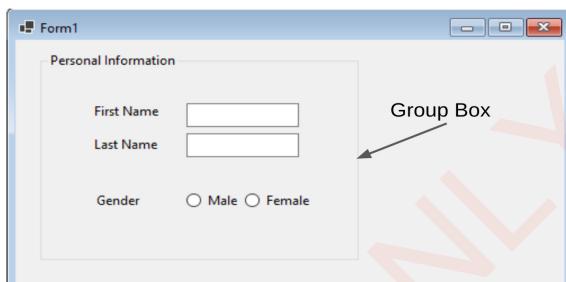
**Figure 4.27:** Container controls in Visual Basic

## GroupBox

The GroupBox control allows you to separate the controls on your form into different groups depending on their functionality. This container is useful since it can provide good structure to forms with many controls. A groupBox allows you to add controls in a frame/box which can be named, making it easier for users to understand which task the controls in the group cover.

To add a GroupBox to your form application, double-click on GroupBox

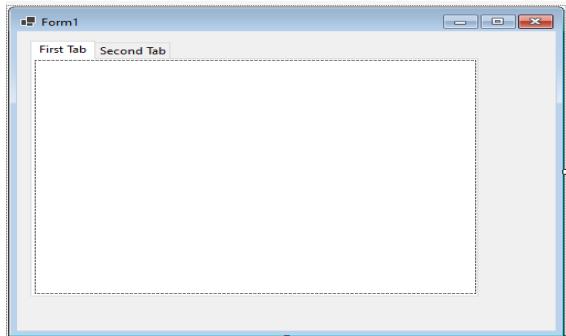
in the ToolBox. Use the properties window to change the name of your GroupBox. To add controls to the GroupBox, select the GroupBox then drag and drop controls to it. Figure 4.28 shows the GroupBox



**Figure 4.28:** Group box titled “Personal Information” containing label, textbox and radiobutton controls

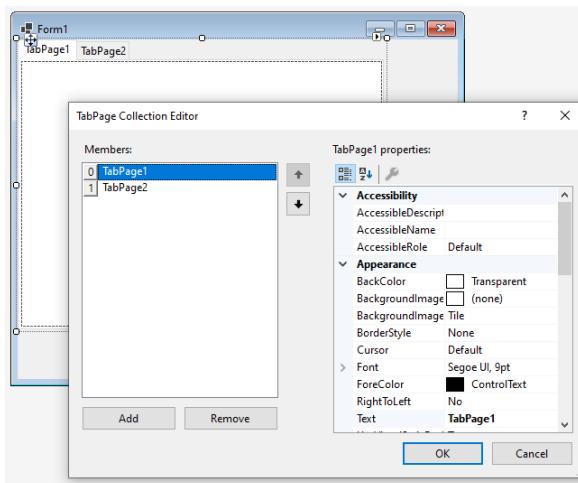
## TabControl

The TabControl allows you to have several Forms in one window using tab style. Each tab on the TabControl can contain different controls. To add a TabControl to your application, double Click on TabControl from the ToolBox as shown in Figure 4.29.



**Figure 4.29:** TabControl example containing two tabs (First tab and Second tab)

The TabControl has many properties that allow you to configure it. To add and remove a tab to the TabControl use the property named TabPages. This will open up a window similar to Figure 4.30, which allows you to add, name and remove tabs. You can also change the order of your tabs using the upward and downward arrows.



**Figure 4.30:** Editing of the TabControl by using properties window to set the appearance and behaviour of each tab

**Activity 4.2:** Explore other container controls available in the containers tab of the Visual Studio Toolbox and write their uses.

### Menus and Toolbars

Menus allow you to add button-like controls that a user can click on to access a certain functionality or task. Menus typically consist of one or two words displayed in a row on top of the application window (see Figure 4.31). Toolbars provide a container to add utilities (tools) that a user can use in the

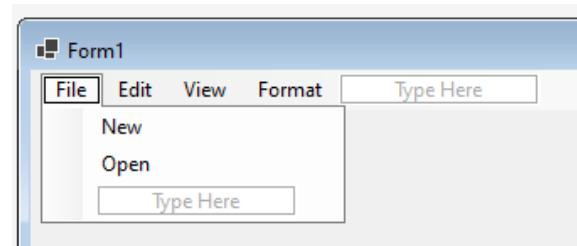
application. An example of a Toolbar is the toolbox window which provides different controls for a programmer. When a menu is clicked it performs a certain task such as launching something. Menus can also be displayed by right-clicking on a control. These are referred to as context menus since they are displayed based on which control is clicked. Visual Studio toolbox has a Menus and Toolbars tab that provides all possible menus and toolbar types.



**Figure 4.31:** Example of a Menu in a Visual Basic form application.

### Adding a menu strip

To add a menu to your form application, double-click on ToolStrip on the toolbox. This will add a row where you can type the name of each menu and sub-menu as shown in Figure 4.32. To provide the functionality of the menu (what should happen if a user clicks on it), double-click on it. This will open the code view where you can write methods which will be called when the menu is clicked.



**Figure 4.32:** Adding Menus and Sub-menus

## FOR ONLINE USE ONLY DO NOT DUPLICATE

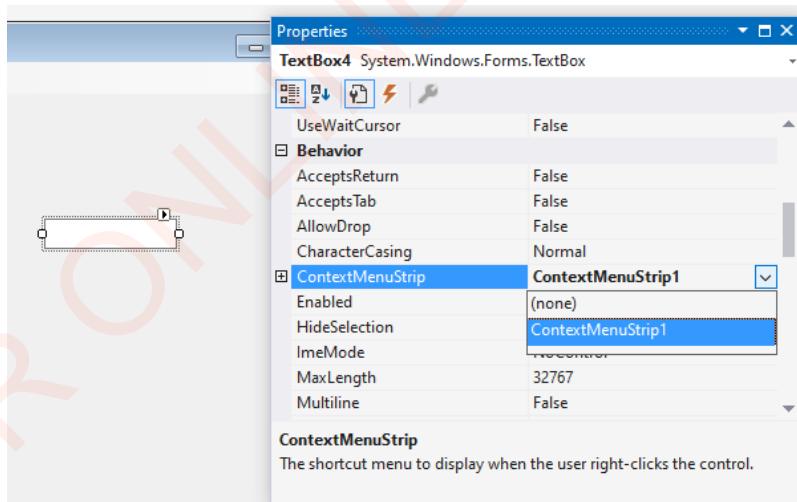
### Adding a context menu

As previously mentioned, context menus are menus that show when a certain control is right-clicked. They are called context menus because they are specific to the control that is clicked. Let us assume that we want to add a context menu that will allow a user to copy, cut or paste text in a TextBox. To achieve this, on your form application drag and drop a TextBox. Then drag and drop a context menu and add three menu items to it (Copy, Cut, Paste). To associate the context menu with the appropriate control, edit the property of the TextBox called *ContextMenuStrip* and select the context menu as shown in Figure 4.33. To add code to the specific menu, double-click on it and in the code view, add the copy, cut and paste methods as shown in Figure 4.33.



**Figure 4.33:** Adding a context menu with three menu items (Copy, Cut and Paste)

Figure 4.34 shows the procedure to associate as specific control with context menu using the properties.



**Figure 4.34:** Associating a specific control (TextBox) with the context menu using the properties window.

## FOR ONLINE USE ONLY

The code snippet in Figure 4.35 activates the copy, cut and paste.

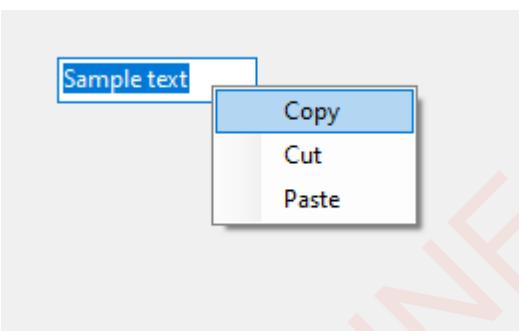
```
0 references
Private Sub CopyToolStripMenuItem_Click(sender As Object, e As EventArgs) Handles
    CopyToolStripMenuItem.Click
    MyTextBox.Copy()
End Sub

0 references
Private Sub CutToolStripMenuItem_Click(sender As Object, e As EventArgs) Handles
    CutToolStripMenuItem.Click
    MyTextBox.Cut()
End Sub

0 references
Private Sub PasteToolStripMenuItem_Click(sender As Object, e As EventArgs) Handles
    PasteToolStripMenuItem.Click
    MyTextBox.Paste()
End Sub
```

**Figure 4.35:** Code to enable the Copy, Cut and Paste context menus to work accordingly.

The output will be as shown in the Figure 4.36.



**Figure 4.36:** Output of the context menu program.



### Activity 4.3: Creating the first program on VB.NET applying basic controls

Assume that you have finished Form VI and want to apply to be admitted to the University of Dar es Salaam. Create a program to display a registration form that captures the following information.

1. First name
2. Last name

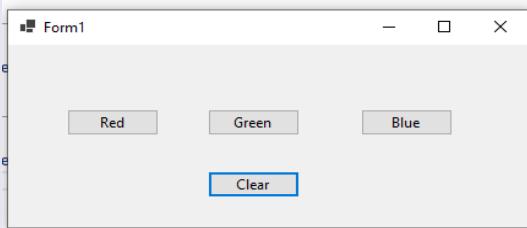
3. Gender
4. Form VI index number
5. Date of form VI completion
6. Select a degree program
7. Add a picture

On the form, add a Button control which when clicked displays the information entered.

### Exercise 4.2

1. Explain the steps used to create a Visual Basic .NET windows forms application project
2. Briefly explain the steps on how to save a new Visual Basic .NET windows forms application project
3. Explain the components found in the Menu bar of the Visual Studio IDE
4. What is the use of a Toolbox in the Visual Studio IDE?
5. Design a form which displays your name and class in two text

- boxes after clicking the buttons “name” and “class” respectively. Add a “clear” button which, when clicked, clears the text boxes.
6. Design a form with three Buttons labelled “red”, “green” and “blue” respectively as shown in Figure 4.37. For each button, write the code to change the Form background colour to the respective colour (red, green or blue) when the button is pressed. Add another button named “clear” which will reset the Form background colour to the default.
- 11P Briefly explain the advantages and disadvantages of VB.NET.
12. Create a form using VB.NET that loads a list of items into a Combo box when the form is loaded.



**Figure 4.37:** A program to set the background colour of the form depending on the button pressed.

7. Describe the following features of Visual Basic studio
- Menu Bar.
  - Toolbox.
  - Form design view.
  - Solution explorer
  - Properties window.
  - Code editor window.
8. Explain the steps for creating menus and drop-down menus in VB.NET.
9. Using VB.Net, create a project that populates a list of items into a list box when the form is loaded.
10. Briefly identify the difference between List box and Combo box

## Programming in Visual Basic .NET

### Data types in Visual Basic

Data types refer to the type of data or value assigned to a variable or function to hold a specific data type. When we declare a variable, we inform the compiler what type of data or value would be assigned to a declared variable for it to allocate enough memory for a value expected to be stored. There are two types of data types in visual Basic .NET: numeric data types and non-numeric data types.

### Numeric data types

Numeric data types consist of numbers, which can be computed mathematically using various standard operators such as addition, subtraction, multiplication, division, and others. Numeric data types include Byte, Integer, Long, Single, Double, Currency and Decimal.

#### Byte data type

This data type requires just one byte of memory. Variables with the Byte data type can store values from 0 to 255. The default byte value is 0. Negative values and values greater than 255 are not allowed.

#### Example 1: Using a variable of type Byte

```
Dim variableName As Byte  
variableName = 5  
...
```

## FOR ONLINE USE ONLY

## DO NOT DUPLICATE

*Integer data type*

The integer data type accepts positive, negative and zero values. Their values range between -32,768 to 32,767. This data type takes up four bytes of memory. One of the most used data types is the integer.

**Example 2:** Using a variable of type Integer

```
Dim vname1, vname2 As Integer
vname1 = 20
...
vname2 = -30
```

*Long data type*

This is an integer data type that stores larger integer values that takes up more memory than a four-byte integer variable. It has a range from -2,147,483,648 to 2,147,483,647 values and is assigned 8 bytes of memory.

**Example 3:** Using a variable of type Long

```
Dim k As Long
k = 2147483648
...
```

*Decimal data type*

This is a numeric data type with a high level of precision. A scaling factor is a difference between the total number of digits and the number of digits to the right of the decimal point. In Visual Basic, numbers are scaled by a power of 10. When working with large numbers that require precision, such as financial calculations, these types are appropriate. This takes up 16 bytes of memory. The values of this data type fall into the following range +/-7.9228162514264337593543950335. With no decimal point, the largest possible number is +/-79,228,162,514,264,337,593,543,950,335 and with 28 decimal points, the largest number is +/-7.9228162514264337593543950335. The smallest non-zero value accepted is +/-0.00000000000000000000000000000001. The syntax is *Dim variableName as Decimal*

**Example 4:** Using a variable of type Decimal

```
Dim DecValue As Decimal
DecValue = 1000.4560323034
...
```

*Double data type*

Double is used to hold double-precision floating-point numbers. It has takes up 8 bytes of memory. Negative values range from -1.79769313486231E308 to -4.94065645841247E-324. For positive values, the range is 4.94065645841247E-324 to 1.79769313486232E308. Just like with the single data type, the double data type is defined using the hash (#) symbol. The syntax is *Dim VariableName As Double* or *Dim VariableName#*

**Example 5:** Using a variable of type Double

```
Dim douValue As Double
Dim douVal1#
douValue = 100.21
douVal1 = 333.44
```

#### Single data type

The Single data type stores floating-point values that do not require the full data width of the Double data type. The range of Single is from -3.4028235E+38 to -1.401298E-45 for negative values and from 1.401298E-45 to 3.4028235E+38 for positive values. The shorthand notation for declaring Single variables is to use a variable name and an exclamation mark (!). The default value of Single is 0. It is also called single precision.

**Example 6:** Using a variable of type Single

```
Dim SingleVal1 As Single
Dim SingleVal2!
SingleVal1 = 123
SingleVal2 = 333.44
```

#### Non-numeric data types

Non-numeric data types are those that cannot be manipulated mathematically. Examples of non-numeric data types are String, Date, Boolean (data types that contain only two values; *true or false*), Object, and Variant. These are described in next subsection.

#### Boolean data type

This data type takes up to two bytes of memory and can only store two values: True or False. To put it another way, the Boolean variable can only have

a value of TRUE or FALSE, or 1 or 0 correspondingly.

A Boolean variable's default value is False. The Syntax is *Dim Vname As Boolean*

**Example 7:** Using a variable of type Boolean

```
Dim testBoolVariable As Boolean
testBoolVariable = True
```

#### String Data Type

A string is a defined collection of characters that have been defined. As a result, the string data type can be used to hold text and digits, special characters, and even spaces. Double quotation marks “ ” are used to enclose string values. There are two types of string data types. Variable-length String and fixed-length String. Variable-length String, which occupies 10 bytes of storage size plus the memory required for the String. They range from 0 to approximately 2 billion characters in length. The fixed-length String is the type of String that occupies the memory equal to the length of the String itself. It can range from 1 to approximately 65,400 characters. The Syntax is *Dim variableName As String OR Dim Vname\$*.

**Example 8:** Using a variable of type String

```
Dim sVal1 As String
sVal1 = "Text1234#@$ and Spaces: )
```

#### Date data type

The date data type is used to represent dates and time. It has date range values from Jan 1, 0100 to December 31, 9999,

and time values from 0:00:00 to 23:59:59 and occupies 8 bytes of memory. The syntax is *Dim variableName As Date*.

**Example 9:** Using a variable of type Date

```
Dim datetime As Date
datetime = Now
```

Note that in the above example, the word “Now” is a function that returns the current date and time.

#### *Object data type*

The Object data type is also referred to as a non-primitive or reference data type. They are so called because they are used to refer to objects of any data type. The Object data type occupies 4 bytes of computer memory. The default value of an object is a null reference indicating that the object has no data. You are advised to use Object when you do not know the data type your variable will point to during runtime.

**Example 10:** Using a variable of type Object

```
Dim wsActiveForm As Object
wsActiveForm = ActiveForm
```

The example above declares a variable with name *wsActiveForm* of type object and then assigns an object called ActiveForm to the declared variable.

#### **Constants**

Constants refer to fixed values that the program will not alter during its execution. These fixed values are also called literals. Constants can be of any basic data type like an integer constant, a floating constant, a character constant, or a string constant. There are also

enumeration constants as well. The constants are treated just like regular variables, except that their values cannot be modified after their definition.

#### **Declaring constants**

The syntax of constant is *Const ConstantName As DataType = Value*

Whereby;

- ConstantName - specifies the name of the constant.
- Datatype - specifies the datatype of the constant
- Value - specifies the value assigned to the constant; the initializer.

**Example 1:** Declaring constants

```
Const Pi As Single = 3.142
Const Name As String = "Name_that _does_not_change"
```

The *const* statement is used at Module, class, structure, procedure, or at the block level for use in place of literal values. The constant may be declared using const statement as follows:

*Const Dim variablename As Datatype.*

**Example 2:** Using constant

```
const Dim pi As Double = 3.142
```

#### **Lvalues and Rvalues**

There are two kinds of expressions which are Lvalues and Rvalues.

*Lvalues:* the expression that is lvalues, it may appear either at the left or at right hand side of an assignment. Variable are lvalues, therefore, they appear on the left hand side of an assignment.

*Rvalues:* an rvalue expression may appear on the right of an assignment but not on the left side. Numeric literals are

rvalues that is why may not be assigned and cannot appear on the left hand side of the assignment.

### Types of constants

Constant can be of any of basic datatype like an integer constant, a floating constant, a character constant or literal constant. There are also an enumeration constant as well. Enumeration is nothing but just a set of named integer constants.

### Visual Basic .NET variables

A variable is a named memory location where a value can be stored throughout the execution of a program. Each variable has a specific type, which determines the following things:

- size and layout of the allocated memory,
- range of values that are allowed to be stored in that memory location
- a set of operations that can be done to the variable

### Rules for naming variables

The basic rules for naming variables are listed as follows:

- It must be less than 255 characters.
- No spacing is allowed.
- It must not begin with a number.
- The period is not permitted.
- They may not include punctuation or other symbols, except for underscore.
- They may contain any mixture of letters and numbers.

Table 4.2 illustrates examples of the valid and invalid variable.

**D**Table 4.2: Example of valid and invalid variable names

| Valid variable name   | Invalid variable name |
|-----------------------|-----------------------|
| My_Name               | My.Name               |
| VB2022                | 2022VB                |
| Long_name_can_be_used | LongName&can&be&used  |

### Declaration of variables

To declare a variable, you need to give it a name and state its data type. Variables are usually declared in the general section of the code windows using the *Dim* statement, and the syntax is *Dim VariableName As DataType*. If you want to declare more variables, you can declare them in separate lines, or you may also combine multiple variables in one line, separating each variable with a comma, for example, *Dim VariableName1 As DataType1, VariableName2 As DataType2, VariableName3 As DataType3*

**Example 1:** The code below shows an example of how to declare variables of different types.

```
Dim MyName As String
Dim Num1 As Integer
Dim Num2 As Single
Dim StartDate As Date
```

### Variable initialization

Variables are initialised (assigned a value) with an equal sign (=) followed by a constant expression. The general form of initialisation is

Variable\_name = value;

### Example 2:

```
Dim pi As Double
pi = 3.14159
```

## FOR ONLINE USE ONLY

Notice that you can assign a value to a variable in one line using the = sign instead of declaring the variable and then giving it a value in another line.

the compiler to conduct specified mathematical or logical operations.

***Dim YourName As String = "Juma"***

is the same as

***Dim YourName As String***  
***YourName = "Juma".***

### ***Types of operators***

The following are commonly used types of operators in Visual Basic:

#### ***Arithmetic operators***

Arithmetic operators are used to perform many of the familiar arithmetic operations that involve the calculation of numeric values represented by literals, variables, other expressions, functions, and constants. Table 4.3 shows arithmetic operators and their examples.

## **Visual Basic operators**

Just like other programming languages, Visual Basic also has operators. Operators are symbols that instruct

**Table 4.3:** Symbols used in arithmetic operators and their applications

| <b>Operator</b> | <b>Description</b>                                                          | <b>Example (a = 5, b = 3)</b> |
|-----------------|-----------------------------------------------------------------------------|-------------------------------|
| +               | Adds two operands                                                           | $a + b = 8$                   |
| -               | Subtracts two operands                                                      | $a - b = 2$                   |
| *               | Multiplies two operands                                                     | $a * b = 15$                  |
| /               | Divides two numbers and returns a floating-point result                     | $a / b = 1.6666$              |
| \               | Divides two numbers and returns an integer as a result without a remainder. | $a \ b = 1$                   |
| Mod             | Divides two numbers and returns only the remainder                          | $a \text{ Mod } b = 2$        |
| ^               | Raises a number to the power of another number                              | $a ^ b = 125$                 |

The following program demonstrate the use of arithmetic operators. The program is presented in Figure 4.38. The program output is presented in Figure 4.39.

```
Public Class Form1
    Dim sum As Integer
    Dim a As Integer
    Dim b As Integer
    Dim product As Integer
    Dim div As Integer
    Dim diff As Integer
    Private Sub Label3_Click(sender As Object, e As EventArgs) Handles Label3.Click
        End Sub

    Private Sub Button3_Click(sender As Object, e As EventArgs) Handles Button3.Click
        a = TextBox1.Text
        b = TextBox2.Text
        product = a * b
        TextBox3.Text = product
    End Sub

    Private Sub Button5_Click(sender As Object, e As EventArgs) Handles Button5.Click
        TextBox1.Clear()
        TextBox2.Clear()
        TextBox3.Clear()
    End Sub

    Private Sub Button1_Click_1(sender As Object, e As EventArgs) Handles Button1.Cli
        a = TextBox1.Text
        b = TextBox2.Text
        sum = a + b
        TextBox3.Text = sum
    End Sub

    Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click
        a = TextBox1.Text
        b = TextBox2.Text
        diff = a - b
        TextBox3.Text = diff
    End Sub

    Private Sub Button4_Click(sender As Object, e As EventArgs) Handles Button4.Click
        a = TextBox1.Text
        b = TextBox2.Text
        div = a / b
        TextBox3.Text = div
    End Sub
End Class
```

Figure 4.38: A program code for arithmetic operators

FOR ONLINE USE ONLY

DO NOT DUPLICATE

|                                                                                       |                                 |
|---------------------------------------------------------------------------------------|---------------------------------|
| Enter the first number                                                                | <input type="text" value="25"/> |
| Enter the second number                                                               | <input type="text" value="5"/>  |
| Result                                                                                | <input type="text" value="30"/> |
| <input type="button" value="Addition"/> <input type="button" value="Subtraction"/>    |                                 |
| <input type="button" value="Multiplication"/> <input type="button" value="Division"/> |                                 |
| <input type="button" value="Reset"/>                                                  |                                 |

**Figure 4.39:** Sample output for arithmetic operations**Exercise 4.3**

1. Write a program that takes in any text and computes its length.
2. Write a program that adds and subtracts two integer numbers. Use two buttons, one for add another for subtract and a message box to display the results.
3. Write a program to compute the area of a triangle. The program should allow a user to input the height and base of the triangle.
4. Write a program to enter the name of a student and three marks, for Physics, Mathematics and Computer Science respectively. The program should compute the average and display the student's name and the average of the three marks in two different labels.
5. Write a program that receives any number, then outputs the square of that number on the screen.
6. Write a program that lets a person enter their date of birth and then computes and displays their age. Hint: Use the *MonthCalendar* control.
7. Write a program that can be used to find the roots of a quadratic expression.  

$$ax^2 + bx + c = 0.$$

*Relational operators*

In Visual Basic, relational operators are used to determine whether the defined two operands are equal, greater than, or less than. Table 4.4 lists the different comparison operators available in Visual Basic.

**Table 4.4:** Symbols used in relational operators and their applications

| Operator   | Description                                                                       | Example (a = 10, b = 5) |
|------------|-----------------------------------------------------------------------------------|-------------------------|
| <          | It returns True if the right operand is greater than the left operand             | a < b : False           |
| <=         | It returns True if the right operand is greater than or equal to the left operand | a <= b : False          |
| >          | It returns True if the left operand is greater than the right operand             | a > b : True            |
| >=         | It returns True if the left operand is greater than or equal to the right operand | a >= b : True           |
| =          | It returns True if operands are equal                                             | a = b : False           |
| $\diamond$ | It returns True if operands are not equal                                         | a $\diamond$ b : True   |
| Is         | It returns True if two object references refer to the same object                 | a Is b : False          |
| IsNot      | It returns True if two object references refer to different objects               | A IsNot b : True        |

*Logical operators*

Logical operators return a Boolean result (True or False) after evaluating one or more boolean expressions. There are six logical operators, which are supported by VB as demonstrated in Table 4.5.

**The concept of short-circuiting**

When using logical operators, short circuiting means that not all conditions in the whole expression are evaluated. The evaluation of the expression ends when the result is already known from only evaluating some of the conditions in the logical expression. In Visual Basic, the AndAlso and OrElse operators perform short circuiting.

Note that, in other object oriented programming languages such as Java and C++, the AND and OR operators perform short circuiting.

Table 4.5 shows logical operators in VB and their applications.

**Table 4.5:** Symbols used in logical operators and their applications

| Operator | Description                                                                                                                                                                                                                                                                                            | Example                                                                                        |
|----------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------|
| AND      | It is the logical AND operator. If both the operands are true, then the condition becomes true. This operator does not perform short-circuiting.                                                                                                                                                       | If A = True and B = True, then (A AND B) = True                                                |
| OR       | It is the logical OR operator. If any of the two operands is true, then the condition becomes true. This operator does not perform short-circuiting.                                                                                                                                                   | If A = True and B = False then (A OR B) = True                                                 |
| NOT      | It is the logical NOT operator. Used to reverse the logical state of its operand. If a condition is true, then the Logical NOT operator will make it false.                                                                                                                                            | If A = True and B = True then NOT(A and B) = False                                             |
| Xor      | It is the logical exclusive OR operator. It returns True if both expressions are true or both expressions are false; otherwise, it returns False. This operator does not perform short-circuiting, it always evaluates both expressions and there is no short-circuiting counterpart of this operator. | If A=True and B=False then (A Xor B) = False<br>If A = False and B = False<br>(A Xor B) = True |
| AndAlso  | The AndAlso operator is very similar to AND operator except that it exhibits short-circuiting behavior. If the first expression evaluates false, the second expression will never be evaluated and the result is false.                                                                                | If A = False and B = True then (A AndAlso B) = False                                           |

|         |                                                                                                                                                                                                                                                     |                                                         |
|---------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------|
| OrElse  | The OrElse operator is very similar to OR expression except that, it performs short circuiting If the first expression evaluates true, the second expression will not be checked because it cannot change the results, and the results will be true | A = True and B = False<br>then (A OrElse B) = True      |
| IsFalse | It determines whether an expression is false                                                                                                                                                                                                        | If A = True and B = True then IsFalse (A and B) = False |
| IsTrue  | It determines whether an expression is true                                                                                                                                                                                                         | If A = True and B = True then IsTrue (A and B) = True   |

## Visual basic control structures

### Exercise 4.4

- Evaluate the following expressions given the following: A = False, B = True,  
C = True, D= False
  - A And (B OR C)
  - A Or (B AND C)
  - A AndAlso (B OR C)
  - B Xor (C AND D)
- Explain the difference between expressions in question 1(a) and question 1(c)

In writing programs, there often arises the need for a program to choose among different courses of execution or to repeat certain computations many times. Consider the problem of counting odd and even numbers in a given interval, say [0, 1000]. A program to solve this problem will need to test each number in the interval to decide whether it is not divisible by two (it is odd). Thus, the program will need two mechanisms to accomplish this task: (1) to decide whether a number is odd or even and count it appropriately (choose a different course of action); and (2) to test each number in the given range (repetition).

Thus, control structures enable programmers to manage flow of their programs. VB.NET offers two kinds of control structure namely decision (selection) and looping (iteration).

## Types of controls structure

There are three types of control structures, which are:

- Sequence control structure
- Decision /selection control structures
- Iteration/ looping control structure

In this part, the last two controls will be discussed.

### Decision /selection control structures

There are two types of decision control structure, which are:

1. If . . . Then statement
2. Select . . . Case statement

#### *The IF statement*

The if statement tests one or more boolean conditions and lets a program take one alternative paths of execution based on the condition's truth value. There are three forms of the If statement: *If...Then*, *If...Then...Else*, and *If...Then...ElseIf...Else*.

#### *The If...Then statement*

This statement is used to test a condition (or conditions) and allow a block of statements to be executed if the condition evaluates to true. It does not offer an alternative block of statements to execute in case the condition evaluates to false. This form of the If statement has the following syntax:

#### If condition Then

statements to execute when *condition* is true

#### End If

### Example 1

As a school system developer of Mwalimu Nyerere Secondary School, write a VB.NET program that will help the academic teacher enter students' mark then the program should display whether the student passed or failed the exam. Assume the pass mark is 50 and above.

#### Solution

In this program, we will need a form with a label, a button, and a text box. The label will be used to hold text that will describe the purpose of the text box; the text box will allow the user to type (input) the mark; and the button will allow the user to command the program to evaluate the provided mark when clicked. Thus, we will take advantage of the button's click to check the entered value for correctness before performing the intended evaluation.

Therefore, when the button is clicked we will do the following tasks: (1) check if the user entered a value; (2) check if the entered value is valid (3) evaluate the value to determine whether the student passed. All of these tasks can be accomplished through the use of the If statement. We create a VB.NET Windows Forms project with a single form. The form will contain the three controls: label, text box and button. The code for the click event of the command button is shown Figure 4.40, while the design of the form is shown in Figure

4.41 (a). Results obtained after running the program with different inputs are shown in Figure 4.41 (b to c). Notice the different course of action the program follows depending on values of the conditions on the If statements.

```

Private Sub Button1_Click(sender As Object, e As EventArgs) Handles ButtonEvaluate.Click
    ' Extract entered the value and remove spaces if present
    Dim markTxt As String = TextBox1.Text.Trim

    ' Check if the user entered a value at all. If not tell the user and stop
    If markTxt = String.Empty Then
        MsgBox("Please enter marks to proceed ...", "Marks App")
        Return
    End If

    ' If the user entered a value check if it is a valid. if not tell the user and stop
    Dim markVal As Short = Val(markTxt)
    If markVal < 0.0 Or markVal > 100.0 Then
        MsgBox("Invalid mark! Must be bewtween 0 and 100 ...", "Marks App")
        Return
    End If

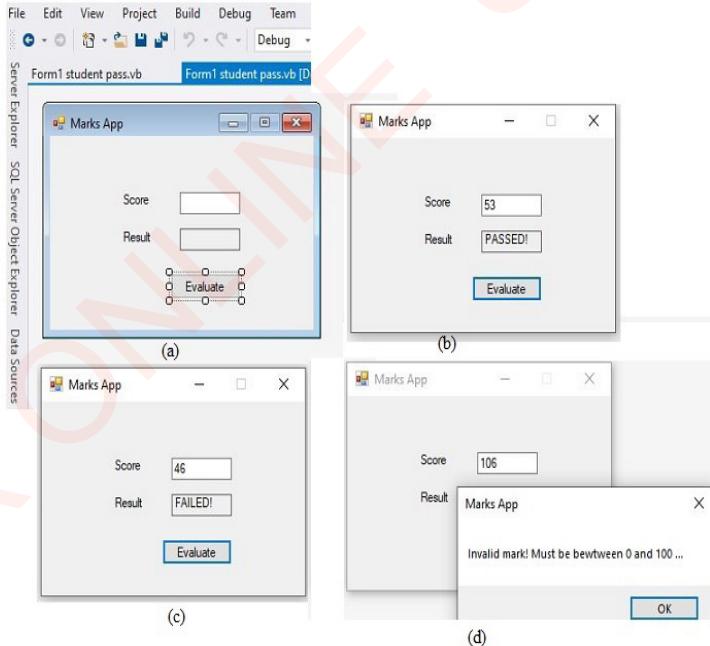
    ' If we get here everything is okay. We can do the evaluation
    If markVal >= 50.0 Then
        Label3.Text = "PASSED!"
        Return
    End If

    If markVal < 50.0 Then
        Label3.Text = "FAILED!"
    End If

End Sub

```

**Figure 4.40:** A program code for evaluating marks as pass or fail



**Figure 4.41:** Design and output of the program: (a) Form design view; (b) Result with a pass mark; (c) Result with a fail mark; and (d) Result with an invalid mark

## FOR ONLINE USE ONLY

*The If...Then...Else statement*  
DO NOT DUPLICATE

Unlike the **If...Else** statement, the **If... Then.. Else** statement tests a condition and allows a program to take one of two courses of action based on the truth value of the condition. That is, when the condition is false it allows an alternative block of statements to be executed. This form of the **If statement** has the following syntax:

```

If condition Then
    statements to execute when condition is true
Else
    statements to execute when condition is false
End If

```

**Example 2**

In example one, two separate If statements were used to test whether the student passed or not. Since passing and failing are mutually exclusive (a student can just either pass or fail), let us improve the program in example one by using the **If... Then...Else** statement.

**Solution**

The only change we need to make is to replace the two if statements which test the value of the *markVal* variable with an **If...Then...Else** statement (Figure 4.42). This change will not affect the behavior of the program but will make it relatively shorter and easy to read. Output of the program is the same as in example 1(Figure 4.41)

```

Private Sub Button1_Click(sender As Object, e As EventArgs) Handles ButtonEvaluate.Click
    ' Extract entered the value and remove spaces if present
    Dim markTxt As String = TextBox1.Text.Trim

    ' Check if the user entered a value at all. If not tell the user and stop
    If markTxt = String.Empty Then
        MsgBox("Please enter marks to proceed ...,, "Marks App")
        Return
    End If

    ' If the user entered a value check if it is a valid. if not tell the user and stop
    Dim markVal As Short = Val(markTxt)
    If markVal < 0.0 Or markVal > 100.0 Then
        MsgBox("Invalid mark! Must be bewtween 0 and 100 ...,, "Marks App")
        Return
    End If

    ' If we get here everything is okay. We can do the evaluation
    If markVal >= 50.0 Then
        Label3.Text = "PASSED!"
    Else
        Label3.Text = "FAILED!"
    End If

End Sub

```

**Figure 4.42:** A program code corresponding to example 2

**Example 3**

Write a program that allows the user to enter a username and display the message “Hello”. If the username entered matches one of the following names “Baraka”, “Neema”, “Ali” or “Issa”, the program should display “Hello” followed by that matching name. Otherwise the program should display the message “Hello, Guest”.

**Solution**

We create a single-form application and put one text box for inputting the user’s name, one button for triggering the greeting, and one label for describing the text box. We implement the functionality of the application in the click event of the button. The code for the click event of the button is shown in Figure 4.43, and the output of the program when run with a known and an unknown user is shown in Figure 4.44. The program uses a dialog box to greet the user.

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    ' Get entered name convert to lower case
    ' ... to simplify comparison
    Dim name As String = TextBox1.Text.ToLower

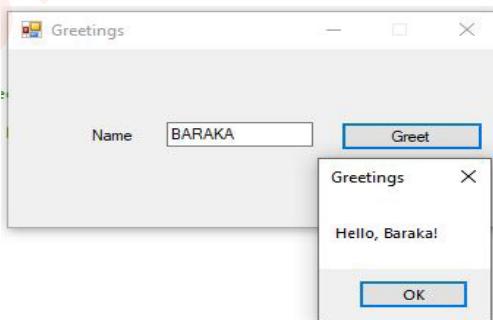
    Dim greeting As String      ' For holding greeting string

    ' Determine whether the user is one of the known users
    ' ... and construct a proper greeting
    If name = "baraka" Then
        greeting = "Hello, Baraka!"
    ElseIf name = "neema" Then
        greeting = "Hello, Neema!"
    ElseIf name = "Ali" Then
        greeting = "Hello, Ali!"
    ElseIf name = "issa" Then
        greeting = "Hello, Issa!"
    Else
        greeting = "Hello, Guest!"
    End If

    MsgBox(greeting, , "Greetings")          ' Display greeting

End Sub
```

**Figure 4.43:** A program code corresponding to example 3



**Figure 4.44:** Output of the program corresponding to example 3

## FOR ONLINE USE ONLY

## TOPIC 5

Note that we convert the name to lower case before we use it in the if statement (line with an arrow) because string comparison is case sensitive. For example, “Neema” and “neema” are treated as different. Thus we convert the entered name to lower case and use lower case comparison in the If statement. Thus, if the user enters “Neema” we convert it to “neema” so that the comparison becomes “neema” = “neema”, which evaluates to true, as opposed to “Neema” = “neema” which evaluates to false.

*The If...Then...ElseIf...Else statement*

This statement is used if there are many conditions to be tested. The conditions are tested one by one from the first to the last and only the block of statements for the first condition that evaluates to true are executed. If none of the conditions evaluates to true, the block of statements in the Else part is executed. This form of the If statement has the following syntax:

**If** *condition1 Then*

statements to execute when *condition1* is true

**ElseIf** *condition2 Then*

statements to execute when *condition2* is true

**ElseIf** *condition3 Then*

statements to execute if *condition3* is true

...

**Else**

statements to be executed if no condition is true

**End If**

*Nested If statement*

The nested If statement is an If statement placed inside another If statement. Nested If statements are often used when you want to test a combination of conditions before deciding on the proper action to take. First, the condition in the outer **If statement** is tested; if it evaluates to true, then the condition in the *inner If* statement is tested. If the condition in the *inner If* statement evaluates to true, the block of statements in the *inner If* statement is executed. The *nested If* statement has the following syntax:

**If** outer condition **Then**

**If** inner condition **Then**

statements to be executed if both conditions are true

**End If**

**End If**

Note that both the outer and inner If statement can have an optional Else part. In this case the statement takes the following form:

**If** outer condition **Then**

**If** inner condition **Then**

statements to be executed if outer and inner conditions are true

**Else**

statements to be executed if outer condition is true but inner condition is false

**End If**

**Else**

statements to be executed if outer condition is false

**End If**

### The Case statement

If you have too many conditions to test, the *If...Then...ElseIf...Else* statement becomes hard to deal with. In this case, the *Select...Case* statement is preferable. The case statement tests a value of an expression and executes one of several Case blocks of statements whose value matches the value of the expression. If the value of the test expression does not match any of the expressions in the case blocks, the

Else part of the statement is executed. The syntax of Case statement is:

**Select Case** test expression

**Case** expression1

block of statements 1

**Case** expression2

block of statements 2

**Case** expression3

block of statements 3

**Case** expression4 ....

**Case Else**

alternative block of statements

**End Case**

### Example 4.

Write a program that prints the name of a day of the week when its corresponding number is entered.

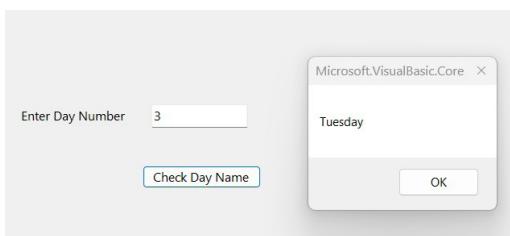
### Solution

Create a new Windows Forms project with a single form. Add a textbox and command button to the form. On the click event of the command button insert the code shown in Figure 4.45.

```
Dim x As Integer
x = Val(TextBox1.Text)
Select Case x
    Case 1
        MsgBox("Sunday")
    Case 2
        MsgBox("Monday")
    Case 3
        MsgBox("Tuesday")
    Case 4
        MsgBox("Wednesday")
    Case 5
        MsgBox("Thursday")
    Case 6
        MsgBox("Friday")
    Case 7
        MsgBox("Saturday")
    Case Else
        MsgBox("Invalid input")
End Select
```

**Figure 4.45:** A program code to display name of a day of the week

Output of the program is shown in Figure 4.46.



**Figure 4.46:** Output of the program corresponding to example 4

### Exercise 4.5

- 1 Write a two-player number guessing game (player1 and player2). The two players play in turns; first, player1 enters a secret number (between 1 and 100) then player2 tries to repeatedly guess the until he or she guesses it correctly. Every time player 2 enters a guess, the game should give feedback as to how far the guess is from the secret number. When the number is guessed correctly, the game should congratulate player 2.
- 2 The body mass index (BMI) is a number used to measure the proportionality between a person's height and weight. For a given height and weight, the BMI tells whether a person is healthy, underweight, overweight, or obese. Furthermore, if a person is obese it tells the severity of the obesity. Write a windows application that calculates BMI of a person and comments on their health status. If the person is obese let the program tell the obesity class as well. Use BMI data in the tables 4.9. Use the formula  $BMI = \frac{\text{weight (kg)}}{(\text{height (m)})^2}$ .

Table 4.6: BMI data and comment and obesity class

| No. | BMI range    | Comment     |
|-----|--------------|-------------|
| 1.  | < 18         | Underweight |
| 2.  | 18.5 to < 25 | Healthy     |
| 3.  | 25 to < 30   | Overweight  |
| 4.  | > 30         | Obese       |

| No. | BMI range  | Obesity class    |
|-----|------------|------------------|
| 1.  | 30 to < 35 | Class 1          |
| 2.  | 35 to < 40 | Class 2          |
| 3.  | > 40       | Class 3 (severe) |

- 3 Write a program that accepts a score and determines a corresponding grade. First, use the If...Then...ElseIf...Else statement, then use the Case statement. Compare the two implementations. Use the following grade table for reference. Table 4.7:

Table 4.7: Examination scores and grades

| No. | Range  | Grade |
|-----|--------|-------|
| 1.  | 75–100 | A     |
| 2.  | 65–74  | B     |
| 3.  | 55–64  | C     |
| 4.  | 45–54  | D     |
| 5.  | 40–44  | E     |
| 6.  | 34–39  | S     |
| 7.  | 0–35   | F     |

### Iteration (Looping)

In the previous section you learnt how to control program flow using the If and Case statements (branching). In this

section, you are going to learn another control structure known as iteration or looping. The iteration control structure enables a program to repeat an action a number of times. Solutions to many programming problems involve some form of repetition, thus, need to repeat actions arises quite often in programming. Looping is achieved using control structures known as *loops*. Like the If statement, loops are driven by Boolean conditions. Some loops repeat an action until a condition becomes true while others repeat an action until a condition becomes false.

### **Uses of iteration control structure**

The iteration is used to repeat line of code or statements or instruction several times in the program.

### **Types of iteration control structures**

VB.NET offers three kinds of loop:

1. While loop
2. Do loop
3. For Each loop

The choice of which type of loop to use depends on several factors including whether you want to repeat some action a fixed number of times; you want to repeat an action until a condition becomes true/false, and; whether you want an action to be performed at least once or not at all.

### **The While Loop**

The while loop repeats a block of statements (the loop body) as long as a given condition is true. The condition is tested before the loop body is entered. Therefore, there is a possibility that the loop body may not be executed at all if the condition is initially false. The entry

condition normally contains a control variable that is altered in each iteration until the condition becomes false. In VB.NET the While loop is available through the *While...End* statement and it has the following syntax:

**While** condition statements  
**End While**

Sometimes you may want to exit a While loop before the loop condition becomes false. You can achieve this by using the *Exit While* statement inside the loop body, usually in an If statement inside the loop body. Similarly, sometimes you may want to skip the current iteration and transfer control to the next iteration of the loop. You can achieve this by using the *Continue While* statement inside the loop body.

### **Example 1**

Suppose you want to write a function that given a positive integer returns the sum of the digits of the integer raised to a given power. For example, calling the function with the number 123 and a power of 2 gives 14.

### **Solution**

To write this function you need to split the number into individual integers. Since you do not know in advance how many digits the number has, you need to repeat the process until a certain condition is met. In this example we use the Mod and the integer division operator (\) operators to successively split the number. We stop when the number becomes zero. The completed function is shown Figure 4.47. The output of the program is shown in Figure 4.48.

FOR ONLINE USE ONLY  
DO NOT DUPLICATE

```

Function SumDigits(ByVal num As Integer, ByVal power As Integer) As Integer
    '
    ' Compute the sum of digits of num raised to power.
    ' Pre condition: num and power are positive integers.
    '

    Dim sum As Integer = 0
    Dim lastDigit As Integer
    While num > 0
        lastDigit = num Mod 10
        sum = sum + lastDigit ^ power
        num = num \ 10
    End While
    Return sum
End Function

```

**Figure 4.47:** A While Loop function to find a sum of integers raised to a given power

Enter a Positive Integer:

Enter Power of the Integer:

Result:

**Calculate Sum of Power of Integers**

**Figure 4.48:** Output a program to find a sum of integers raised to a given power

### The Do Loop

The Do loop repeats a block of statements as long as a given condition is true or until the condition becomes true. It can be tested either before entering the loop or at the end of the loop. Similar to the While loop, if you test the condition before entering the loop and the condition evaluates to false the loop body will not be executed at all. If you test the condition at the end of a loop, the loop body is guaranteed to be executed at least once. Alike to the While loop, the loop body alters a control variable to ultimately make the loop stop. The *Do* loop is available through the *Do...Loop* statement and has the following syntax.

|                                                                 |                                                                         |
|-----------------------------------------------------------------|-------------------------------------------------------------------------|
| <b>Do While/Until</b><br>condition<br>statements<br><b>Loop</b> | <b>Do</b><br>statements<br><b>Loop While/</b><br><b>Until</b> condition |
|-----------------------------------------------------------------|-------------------------------------------------------------------------|

As with the While loop, sometimes you may want to exit a Do loop before the loop condition becomes false/true. You can achieve this by using the *Exit Do* statement inside the loop body, usually in an If statement inside the loop body. Similarly, sometimes you may want to skip the current iteration and transfer control to the next iteration of the loop. You can achieve this by using the *Continue Do* statement inside the loop body.

### Example 2

An Armstrong number of order N is an N-digit number that has the property that the sum of its digits raised to the N<sup>th</sup> power equals the number itself. For example, 153 is an Armstrong number of order 3 since  $153 = 1^3 + 5^3 + 3^3 = 1 + 125 + 27 = 153$ . Write a function that finds the first Armstrong number of order 4.

### Solution

To do this we will need to split numbers. For this purpose, we can make use of the SumDigits function in example 18. The completed function is shown in Figure 4.49. The output of the program is shown in Figure 4.50

```
Function FindFirstArmstrong() As Integer
    ' Find the first order 4 Armstrong number
    ' ...
    Dim startNumber As Integer = 1000
    Dim found As Boolean = False
    Dim nextNumber As Integer = startNumber

    Do
        If SumDigits(nextNumber, 4) = nextNumber Then
            found = True
        End If
        nextNumber = nextNumber + 1
    Loop Until found = True
    Return nextNumber - 1
End Function
```

**Figure 4.49:** A Do Loop function to find the first Armstrong number

Enter a number: 2000

Result: 1634

Calculate Armstrong Number

**Figure 4.50:** Output a program to find the first Armstrong number

### The For Each Loop

The *For Each* loop is used with collections where it runs a block of statements once for each element of the collection. To use the For Each loop you specify a loop control variable which will be assigned a value from the collection in each iteration. The loop exits when all elements in the collection have been processed. The For Each loop is available through the For Each...Next statement and has the following syntax.

|                                         |
|-----------------------------------------|
| <b>For Each</b> element <b>In</b> group |
| statements                              |
| <b>Next</b>                             |

## FOR ONLINE USE ONLY

## FOR ONLINE USE ONLY

To exit a For Each loop prematurely or to skip the current loop iteration you can use the *Exit For* and *Continue For* statements, respectively based on evaluation of a condition in an If statement.

**Example 3**

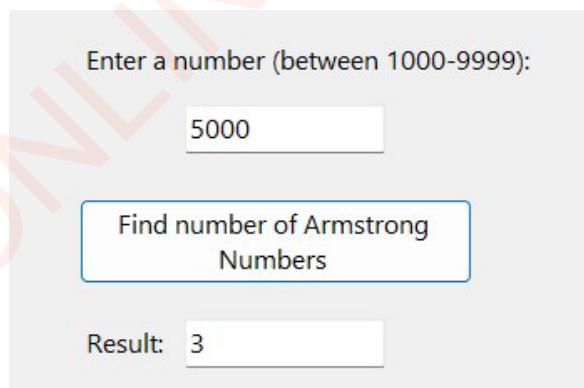
Write a function that return the number of order 4 Armstrong numbers.

**Solution**

In this case the number of iterations is known in advance (testing the numbers between 1000 and 9999), hence we can make use of the For Each loop. Again, we make use of the SumDigits function in example 18. The completed function is shown in Figure 4.51. The output of the program is shown in Figure 4.52

```
Function CountOrder4Armstrong() As Integer
    ' Counts 4th order Armstrong numbers
    '
    Dim startNumber As Integer = 1000 ' first 4-digit number
    Dim numbersToTest = 8999           ' 1000 + 8999 = 9999, last 4-digit number
    Dim count As Integer = 0
    Dim NUMBERS As IEnumerable(Of Integer) = Enumerable.Range(startNumber, numbersToTest)
    '
    For Each number In NUMBERS
        If SumDigits(number, 4) = number Then
            count = count + 1
        End If
    Next
    Return count
End Function
```

**Figure 4.51:** A For Each Loop function to find the count of Armstrong numbers



**Figure 4.52:** Output a program to find the count of Armstrong numbers

As with the If statement, you can put one loop inside another. When you do this, the loops are said to be nested. You can nest any type of loop inside another. When loops are nested, for each iteration of the outer loop the inner loop runs to the end.

**Exercise 4.6**

1. Write a program to find the factorial of a number.
2. Construct a program to enter  $n$  numbers (both positive and negative) and find the average of the negative numbers only.
3. Write a program to print multiples of 6 which lie between 12 and 60, using a do while first, then use do loop.
4. Write a program to display odd numbers which lie in a range defined by a user.
5. Write a program to find multiples of 3 that exist between 1 and 100.
6. Write a program to list all Armstrong numbers between 1 and 10000.

**Visual Basic .NET procedures and functions****Meaning of procedures**

A procedure is a block of Visual Basic statements enclosed by a declaration statement (function, Sub, Operator, Get, and Set) and a matching end declaration. All executable statements in Visual Basic must be within some procedures. A function procedure is a series of Visual Basic statements enclosed by the Function and End function statements. The function performs a task and then returns control to the calling function. When it returns control to the calling function, it may also return a value. The syntax of a function is *FunctionName (arguments)*. There are two types of functions; Built-in functions and User defined functions as explained in the following subsection.

**Functions****Meaning of functions**

The function is a separate group of codes that are used to perform a specific task when the defined function is called in a program. Functions are like sub procedures, except they return a value to the calling procedure. Functions are

useful for taking one or more pieces of data, called arguments and performing some tasks with them. Then the function returns a value that indicates the results of the tasks complete within the function. After the execution of a function, control transfer to the *main()* method for further execution.

**Syntax**

```
Private Function
procedureName([parameterList]) As
dataType
Statements
Return expression
End Function
```

The statement's syntax is *Return* expression, where *expression* represents the one and only value that will be returned to the statement invoking the function. The *data type* of the *expression* must agree with the data type specified in the *As dataType* section of the function's header.

**Types of functions****Built-in functions**

In this lesson, you will learn two very basic useful internal functions of Visual Basic the *MsgBox ()* and *InputBox ()* functions

**MsgBox () function**

The objective of MsgBox is to produce a pop-up message box that prompts the user to click on a command button before they can continue. The syntax is `yourMsg=MsgBox (Prompt, Style Value, Title)`.

The first argument, Prompt, will display the message in the message box. The Style Value will determine what type of command buttons appear on the message box; refer to Table 4.8 for the types of command buttons displayed. The Title argument will display the title of the message board.

**Table 4.8:** Style Values

| <b>Style Value</b> | <b>Named Constant</b> | <b>Title of the message</b>     |
|--------------------|-----------------------|---------------------------------|
| 0                  | vbOkOnly              | Ok button                       |
| 1                  | vbOkCancel            | Ok and Cancel buttons           |
| 2                  | vbAbortRetryIgnore    | Abort, Retry and Ignore buttons |
| 3                  | vbYesNoCancel         | Yes, No and Cancel buttons      |
| 4                  | vbYesNo               | Yes and No buttons              |
| 5                  | vbRetryCancel         | Retry and Cancel buttons        |

We can use named constants instead of integers for the second argument to make the programs more legible. In reality, VB.NET will present you with a selection of name constants from which you can choose one.

For example, the program code:

`YourMsg=MsgBox ("Click OK to Proceed", 1, "Startup Menu")` and `yourMsg=MsgBox ("Click OK to Proceed", vbOkCancel, "Startup Menu")` are the same. YourMsg is a variable that holds values returned by the MsgBox ( ) function. The type of buttons the users click determines the values. It has to be set as an Integer data type in the general declaration section procedure. Table 4.9 shows the values, the corresponding named constant and buttons.

**Table 4.9:** The values, corresponding named constant and buttons

| <b>Value</b> | <b>Named Constant</b> | <b>Button Clicked</b> |
|--------------|-----------------------|-----------------------|
| 1            | VbOk                  | Ok button             |
| 2            | vbCancel              | Cancel button         |
| 3            | vbAbort               | Abort button          |
| 4            | vbRetry               | Retry button          |
| 5            | vbIgnore              | Ignore button         |
| 6            | vbYes                 | Yes button            |
| 7            | vbNo                  | No button             |

*InputBox () function*

An InputBox ( ) function will display a message box where the user can enter a value or a message in the form of text. The syntax is *myMessage=InputBox (Prompt, Title, default text, x-position, y-position)*. *myMessage* is a variant data type. Still, typically, it is declared as String, which accepts the users' message input. The description of the arguments is as follows:

**Prompt** - The message is displayed normally in the form of a question.

**Title** - The title of the InputBox.

**Default-text** - The default text that appears in the input field where users can use it as their intended input, or they may change to the message the user wishes to key in x-position and y-position - the

position or coordinate of the InputBox.

**User-defined procedures**

User-defined procedures are procedures that the programmer defines. When you define a procedure, you can give it a name of your choice, but it should not be similar to the built-in function names.

An argument can be passed by value or reference to a procedure. The mode of passing argument determines whether the procedure can change the argument's value in the calling code.

Passing argument by reference allows a called function procedure to return a value to the calling procedure, and the underlying elements can be modified. On the other hand, passing an argument by value, the underlying elements cannot be altered.



#### Activity 4.4: Passing parameters by value and by reference in a user defined procedure

Create a new Windows Form project with a single form and add a button on the form. Open the form's class code and add the two subroutines shown in Figure 4.53. Open the button's code for the click event and add the code shown in Figure 4.54. Run the application and study its behavior.

```
1 reference
Private Sub RaiseToPowerByVal(ByVal num, ByVal power)
    num = num ^ power
End Sub

1 reference
Private Sub RaiseToPowerByRef(ByRef num, ByVal power)
    num = num ^ power
End Sub
```

Figure 4.53: The subroutines for passing parameters.

```

Dim x As Integer = 10
Dim power As Integer = 2

' Display the value of x before calling RaiseToPowerByVal
MsgBox("Value of x before passing to RaiseToPowerByVal: " + x.ToString())
' Call RaiseToPowerByVal
RaiseToPowerByVal(x, power)
' Display the value of x after calling RaiseToPowerByVal: x does not change
MsgBox("Value of x after passing to RaiseToPowerByVal: " + x.ToString())

' Display the value of x before calling RaiseToPowerByRef: x is still 10
MsgBox("Value of x before passing to RaiseToPowerByRef: " + x.ToString())
' Call RaiseToPowerByRef to change x
RaiseToPowerByRef(x, power)
' Display the value of x after calling RaiseToPowerByRef: x changed to x ^ power
MsgBox("Value of x after passing to RaiseToPowerByRef: " + x.ToString())

```

**Figure 4.54:** A program calling the subroutines.

### Activity 4.5: Creating a program using a function to classify entered number

Write a function to classify any entered number according to its sign and display the phrase (negative number) when the number is negative, and the phrase (positive number) when the number is positive, otherwise, display the phrase (neither positive nor negative).

**Program codes are as follows:**

```

1 Dim x As Single
2 Private Sub command1_click ()
3 x = Val (Text1.Text)
4 If x > 0 Then
5 MsgBox "positive number"
6 ElseIf x < 0 Then
7 MsgBox "negative number"
8 Else
9 MsgBox "neither positive nor negative"
10 End If
11 End Sub

```

**Figure 4.55:** A program calling the subroutines.

**Exercise 4.7**

1. Describe the following functions in brief, the Exp(), Round(), Asc(), Right() and Sqrt().
2. Describe how parameters can be passed.
3. In how many ways a function can return value in VB.NET?
4. Describe the user-defined functions with examples.
5. Describe the procedure in VB.NET

### **Linking Visual Basic .NET form to Data source**

It is common for applications to require persistent storage of data. This can be achieved through the use of files and databases. The former is mostly used for persistent storage of configuration data while latter is used for more elaborate data storage needs. This describes the how to work with databases in VB.NET.

The .NET framework provides facilities for working with different databases. These facilities are called data providers and are used to connect to a database (data source), and running queries (executing commands) for retrieving and modifying data in a data source. The retrieved results can be processed by the application or be shown to the user.

Data providers in .NET are made up of four core objects as described in Table 4.10.

**Table 4.10:** Core components of .NET data providers

| No. | Object      | Description                                                                        |
|-----|-------------|------------------------------------------------------------------------------------|
| 1.  | Connection  | Used for establishing a connection to a specific data source, e.g., MS SQL Server. |
| 2.  | Command     | Used for executing commands against a data source.                                 |
| 3.  | DataReader  | Used for reading a forward-only and read-only stream of data from a data source.   |
| 4.  | DataAdapter | Used for retrieving and saving data.                                               |

The .Net framework provides several data provider for use when working with databases in VB.NET applications, these are data providers for: SQL server, Object Linking and Embedding (OLE) DB, Open Database Connectivity (ODBC), and Oracle. The choice of a data provider to use depends on the data source you are planning to use in your application as well as the nature of your application (middle- or single-tier). For example, the ODBC provider is recommended for middle- and single-tier applications.

### **Using the ODBC data provider in a VB.NET application**

This section describes how to use the ODBC data provider to work with an MS Access database. To do this we will

need an existing MS Access database to work with. Complete Activity 4.6 to create a simple Access database for this purpose.



### Activity 4.6: Creating an employee database in MS Access

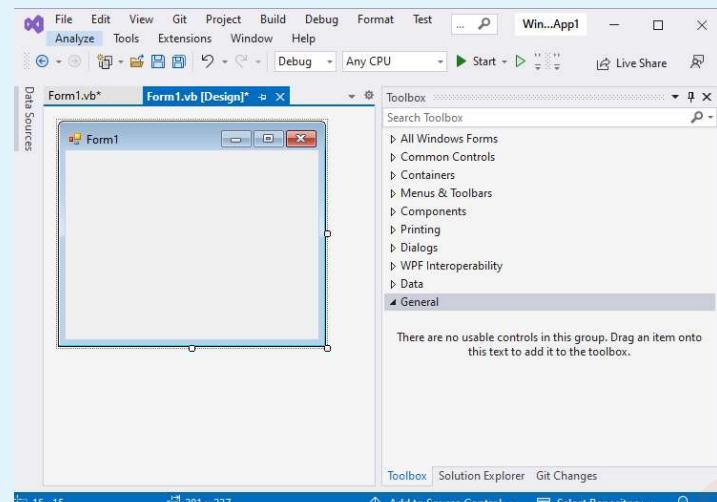
Create an Access database with one table for tracking students' contact details (phone number and email address). Name the table *Contacts* and add the following columns: *firstName*, *middleName*, *lastName*, *gender*, *phoneNumber*, and *emailAddress*. Use the default ID field as primary key. Save the database using the name *employees.accdb*. The finished database should look similar to Figure 4.56.

| ID | firstName | middleName | lastName  | gender | phoneNumber  | emailAddress            |
|----|-----------|------------|-----------|--------|--------------|-------------------------|
| 1  | Jamila    | Selemani   | Mfaume    | Female | 078456892345 | j.mfaume@tie.go.tz      |
| 2  | Josephine | Japhet     | Samson    | Male   | 071445963214 | j.somson@tie.go.tz      |
| 3  | Baraka    | Boniphace  | Mlimwa    | Male   | 062100584769 | b.mlimwa@yahoo.com      |
| 4  | Khalfan   | Issa       | Mohamed   | Male   | 077322348574 | k.mohamed@tie.go.tz     |
| 5  | Kokubanza | Daniel     | Kamugisha | Female | 077312478536 | k.kamugisha@tie.go.tz   |
| 6  | Salma     | Hassan     | Hussein   | Female | 066733568749 | s.hussein@tie.go.tz     |
| 7  | Ayoub     | Jumaa      | Kobelio   | Male   | 075489220103 | a.kobelio@gmail.com     |
| 8  | Mwasiti   | Kobelio    | Mkude     | Female | 078901036457 | mwenda_mkude@gmail.com  |
| 9  | Sofia     | Paul       | Kavakule  | Female | 065588200040 | sofi_kavakule@yahoo.com |
| 10 | Hashimu   | Shabani    | Mwakyosa  | Male   | 076352871001 | h.mwakyosa@tie.go.tz    |

Figure 4.56: Sample database

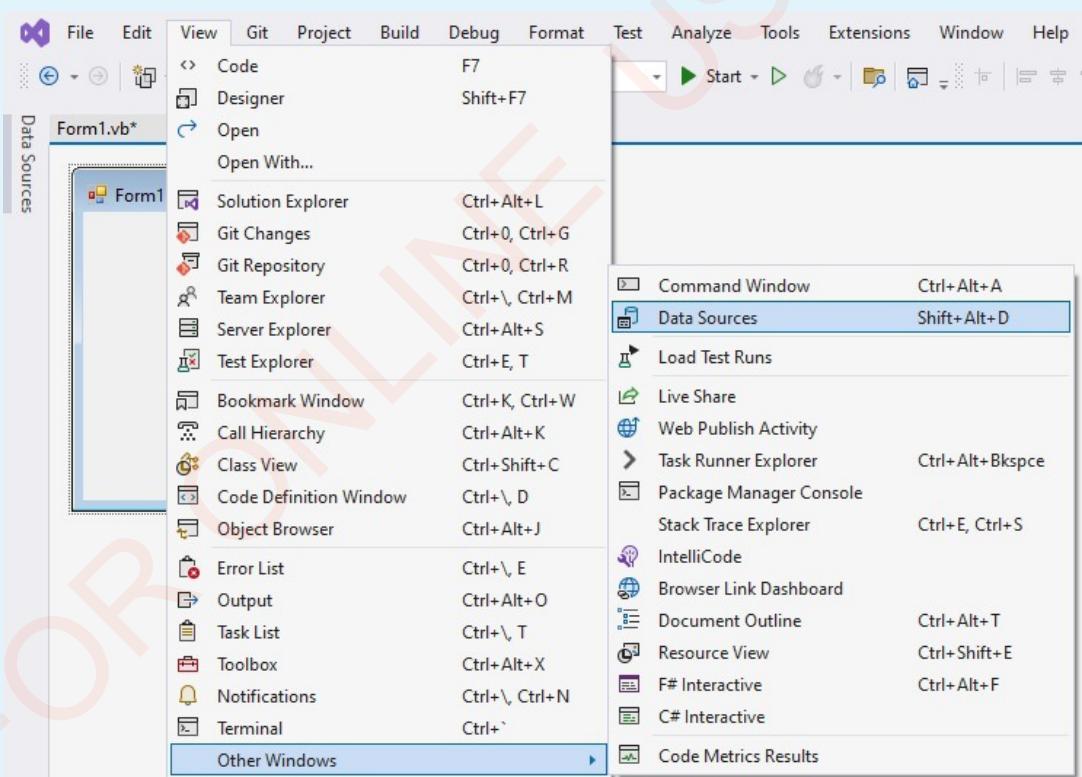
The next step after creating a test database is to create a new Windows Forms project in Visual Studio and add connect to the database. After connecting to the database the data will appear in the Data Sources window where we can drag the table into a form. The following steps takes you on this process.

1. Create a new Windows Forms project with a single form as shown in Figure 4.57.



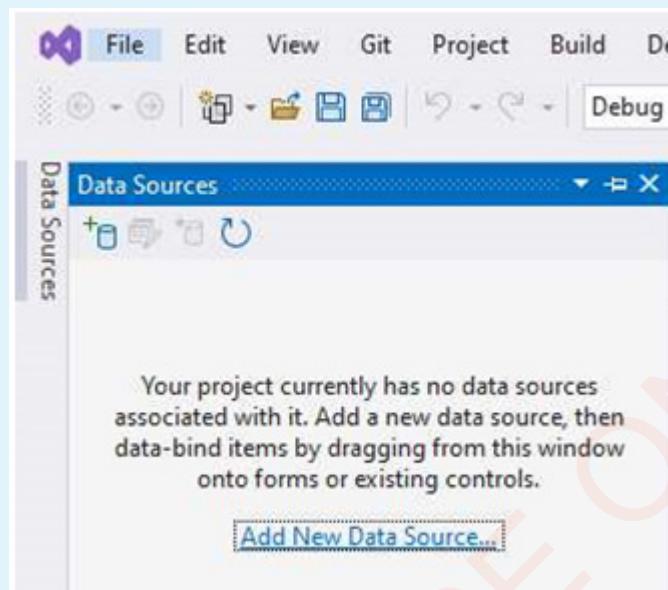
**Figure 4.57: A new Windows Forms**

2. Select **View > Other Windows > Data Sources** to open the **Data Sources** window as shown in Figure 4.58.



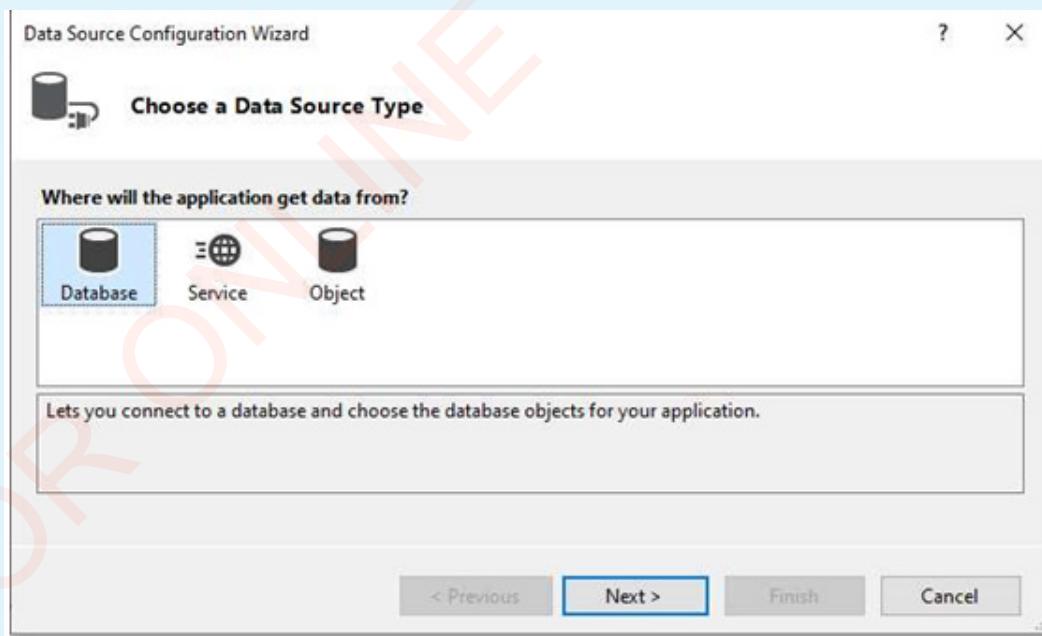
**Figure 4.58: Accessing the Data Sources menu item**

3. On the **Data Sources** window that appears, click **Add New Data Source**. The **Data Source Configuration Wizard** will open (see Figure 4.59).



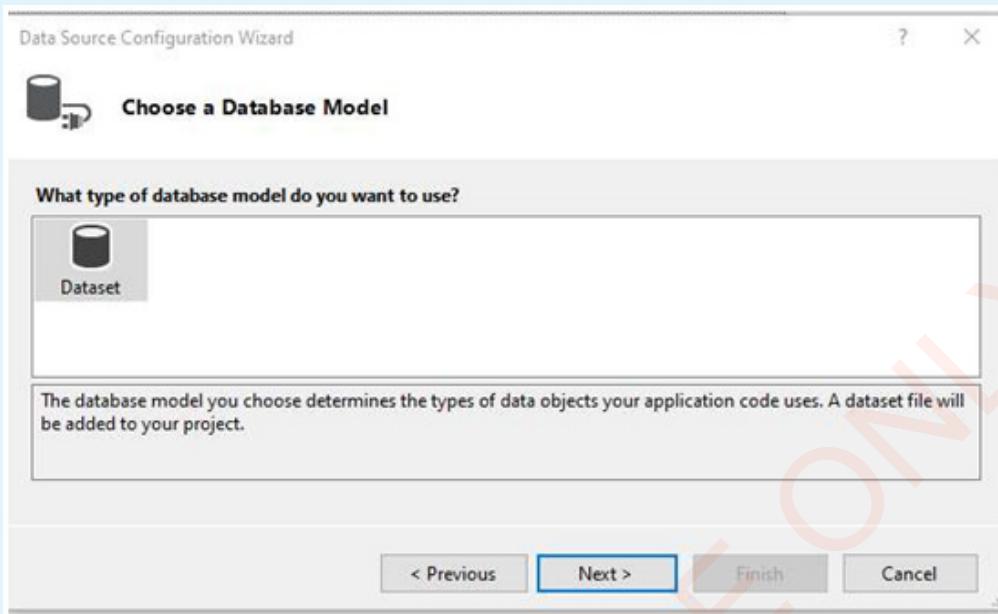
**Figure 4.59:** Adding a new data source to the project

4. On the **Choose a Data Source Type** page select **Database** then click **Next** as shown in Figure 4.52.



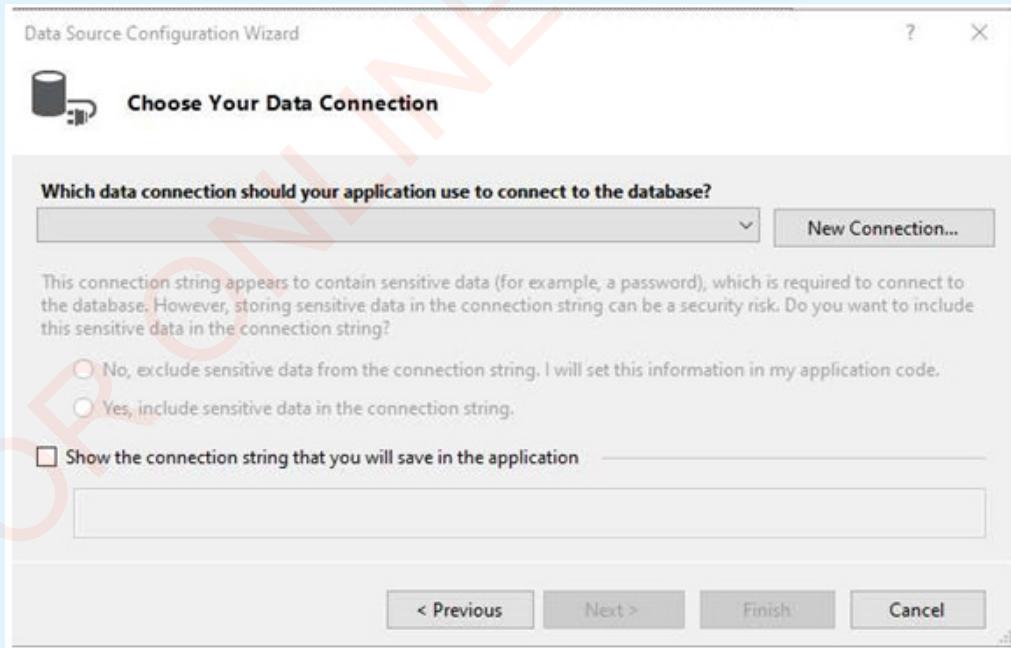
**Figure 4.60:** Choosing data source type

5. On the **Choose a Database Model** page select **Dataset** then click **Next** (Figure 4.61).



**Figure 4.61:** Choosing a data set model

6. On the **Choose your Data Connection** page, select **New Connection** to configure a new data connection (Figure 4.62). The **Add Connection** dialog box will open (Figure 4.63).



**Figure 4.62:** Creating a new connection

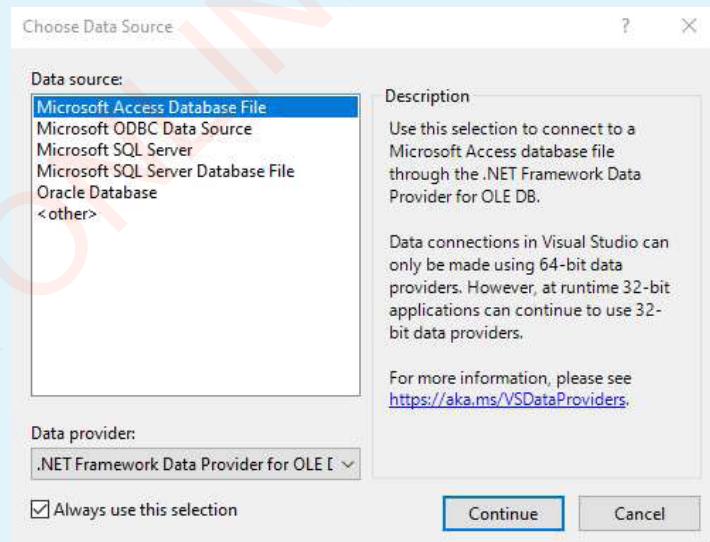
FOR ONLINE USE ONLY

DO NOT DUPLICATE



**Figure 4.63:** A dialog box for adding connection

- If Data source is not set to **Microsoft Access Database File**, select the **Change** button to change it to **Microsoft Access Database File (OLE DB)**. Select **Microsoft Access Database File** then click **Continue** (Figure 4.64).



**Figure 4. 64:** Selecting a data source

8. Choose **Browse** (Figure 4.63) to locate the *employees.accdb* file (Figure 4.65) then click **Open**.

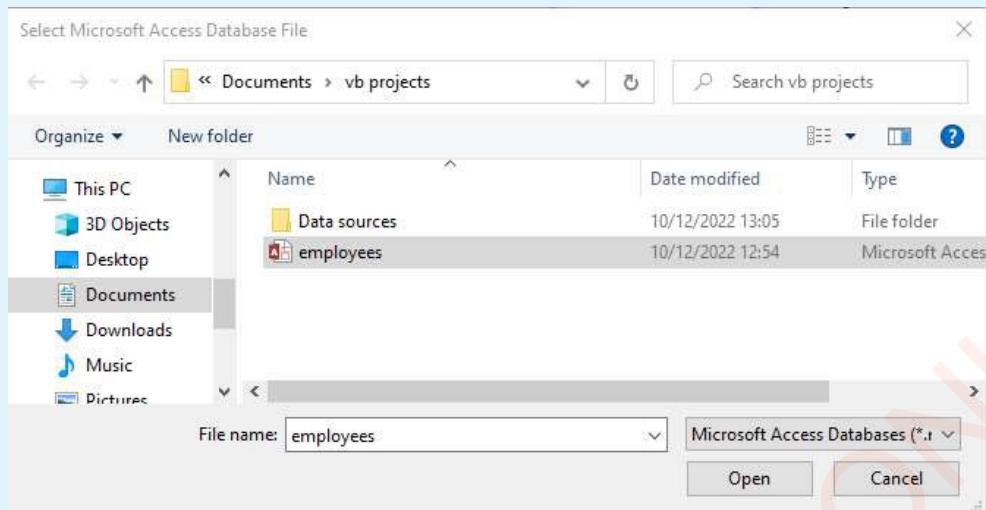


Figure 4.65: Locating a database file

9. Click **Test Connection** ( Figure 4.66) to test connectivity to the database.

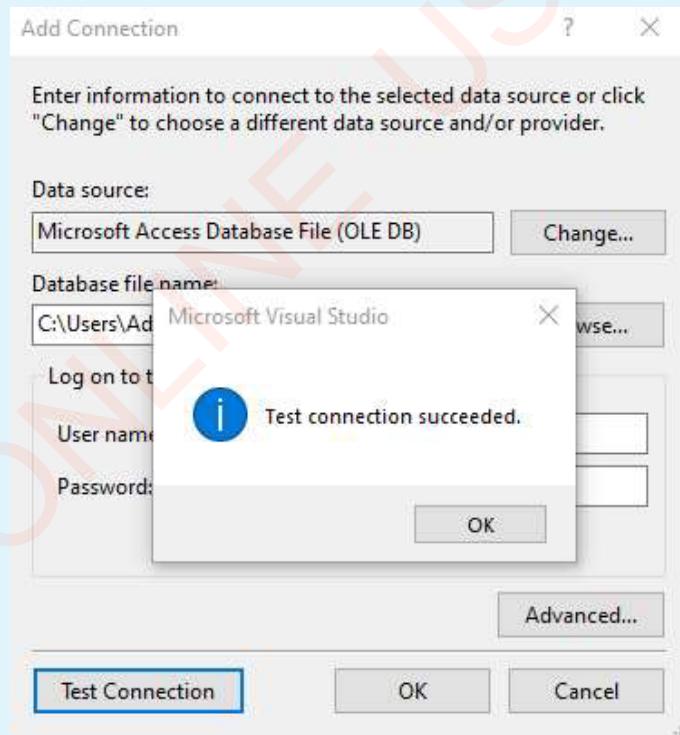
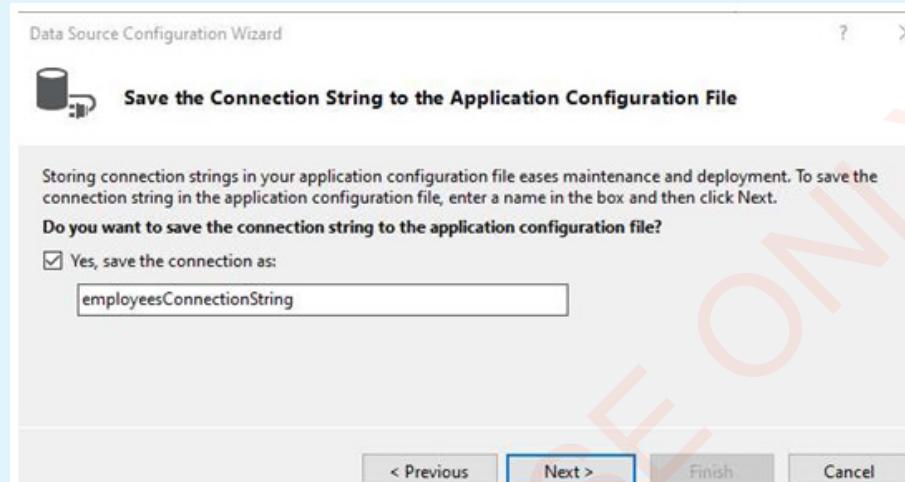


Figure 4.66: Testing a connection to a database

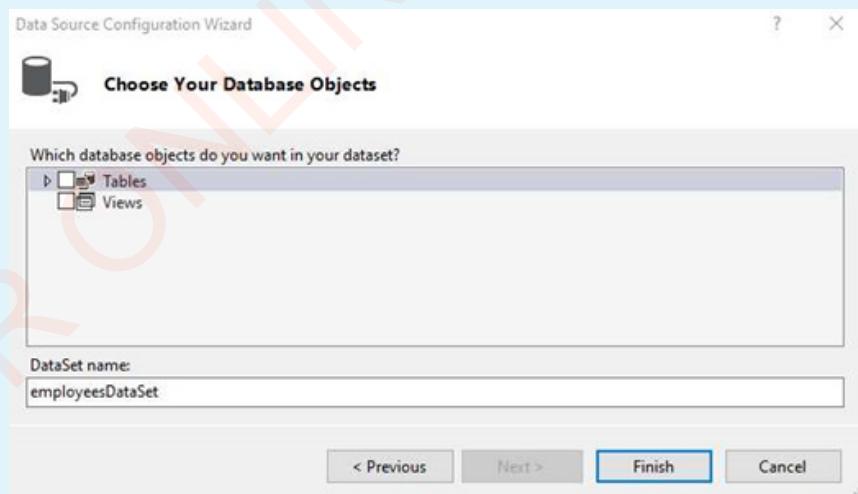
## FOR ONLINE USE ONLY

10. If the connection fails, recheck to make sure you performed previous steps correctly. Click **OK** to close the test result dialog.
11. Click **OK** on the **Add Connection** dialog box.
12. Click **Next** on the **Choose Your Data Connection** page. The **Save connection String to the Application Configuration File** dialog will be displayed (Figure 4.67).



**Figure 4.67:** The Save Connection String to the Application Configuration File dialog

Click **Next** on the **Save Connection String to the Application Configuration file** dialog. The **Choose Your Database Objects** window will be displayed (Figure 4.68).



**Figure 4.68:** Choosing database objects

13. Expand the **Tables** node on the **Choose your Database Objects** page and select the Contacts table to include it in the dataset then click **Finish** (Figure 4.69). The table will appear in the **Data Sources** window as shown in Figure 4.70.

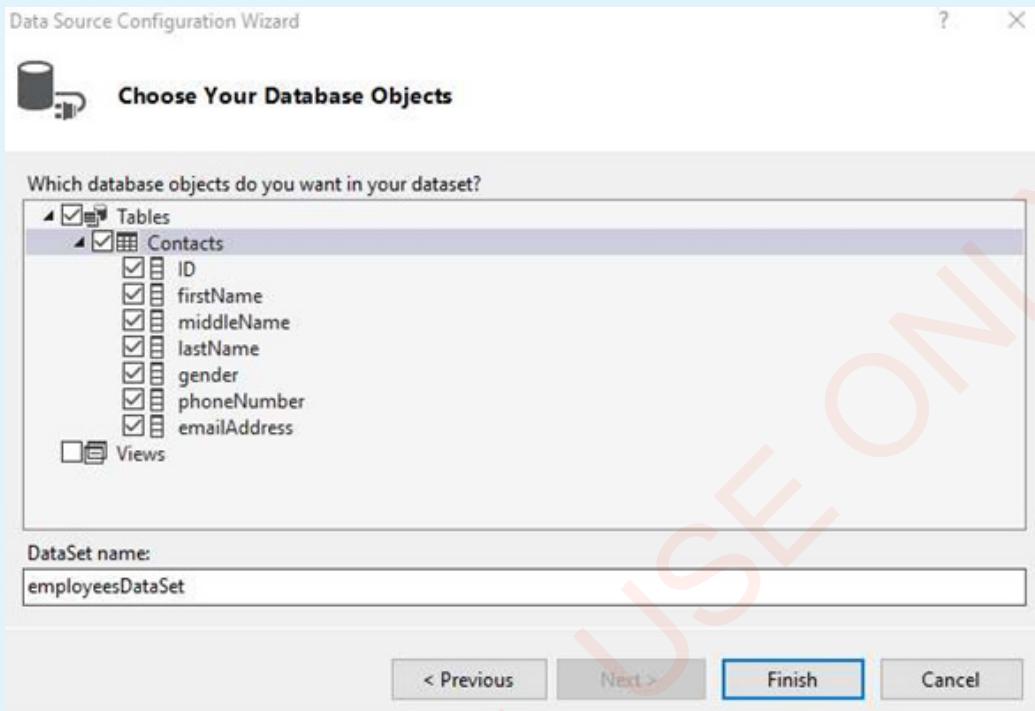


Figure 4.69: An expanded form of a table Contacts

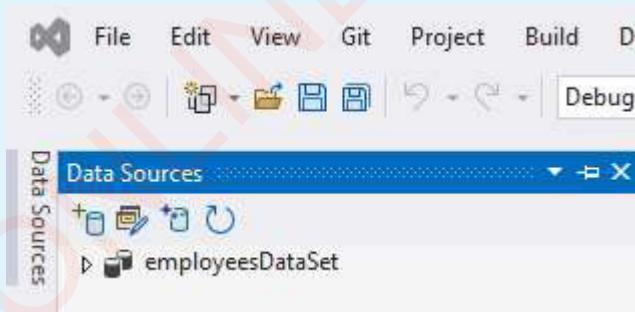


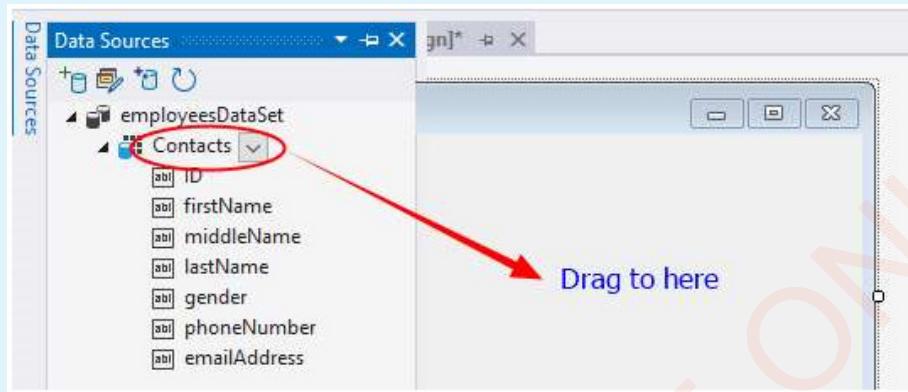
Figure 4.70: The table Contacts appearing in Data Sources window

## FOR ONLINE USE ONLY

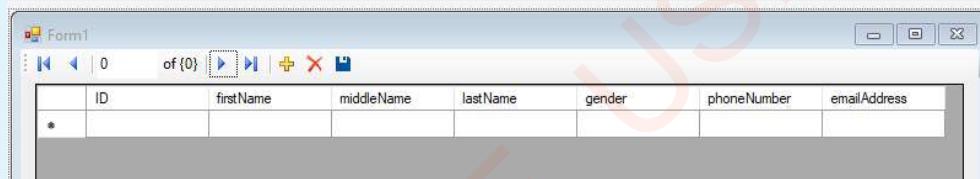
The next step after the data source is created is to add controls on the form for viewing, adding and updating the data in the database. To do this, follow the following steps.

1. Drag the Contacts table from the Data Sources window and drop it on the form (Figure 4.71).

A data grid with all the fields in the Contacts table will appear on the form (Figure 4.72). You may need to resize the grid to make all fields visible.



**Figure 4.71:** Dragging table Contacts and dropping it on the form



**Figure 4.72:** A data grid with all the fields from the Contacts table

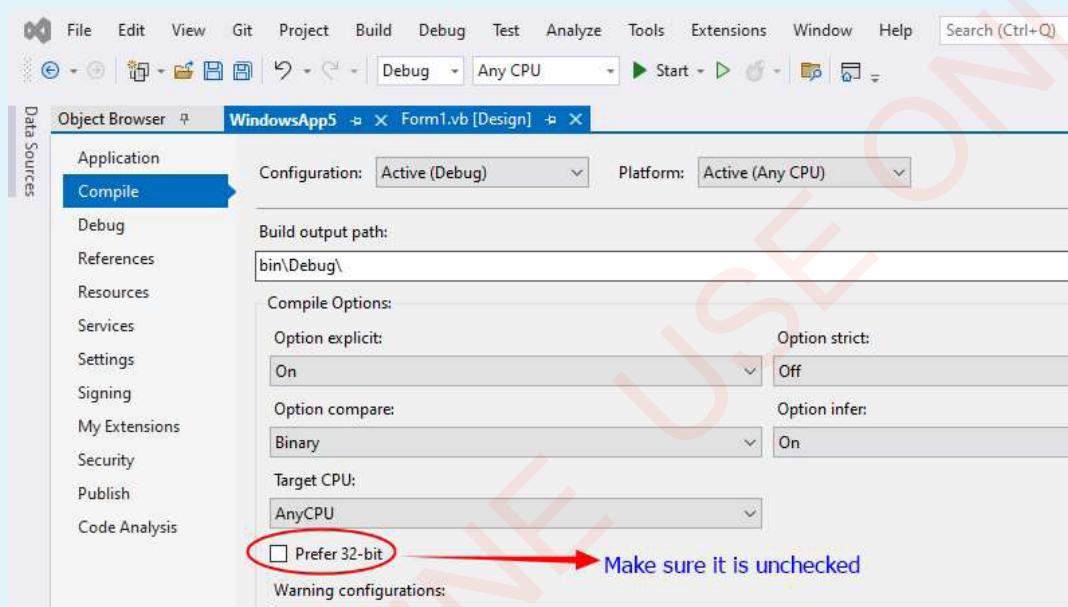
2. Run the application. The data in the Contacts table will be shown in the grid as in Figure 4.73.

| ID | firstName | middleName | lastName  | gender | phoneNumber  | emailAddress       |
|----|-----------|------------|-----------|--------|--------------|--------------------|
| 1  | Jamila    | Selamari   | Mfaume    | Female | 078456892345 | j.mfaume@tie.go.tz |
| 2  | Josephine | Japhet     | Samson    | Male   | 071445963214 | j.somson@tie.go.tz |
| 3  | Baraka    | Boniphace  | Mlimwa    | Male   | 062100584769 | b.mlimwa@yahoo...  |
| 4  | Khalfan   | Issa       | Mohamed   | Male   | 077322348574 | k.mohamed@tie....  |
| 5  | Kokubanza | Daniel     | Kamugisha | Female | 077312478536 | k.kamugisha@tie... |
| 6  | Salma     | Hassan     | Hussein   | Female | 066733568749 | s.hussein@tie.g... |
| 7  | Ayoub     | Jumaa      | Kobelo    | Male   | 075489220103 | a.kobelo@gmail.... |
| 8  | Mwasiti   | Kobelo     | Mkude     | Female | 078901036457 | mwenda_mkude...    |

**Figure 4.73:** Data from the Contacts table displayed on the form

Note that when you run the application you may get an error that the OLE DB provider is not registered on the local machine. When this happens:

7. If you are using VS 2022, make sure you use a 64-bit version of MS Access. If you are using an earlier version of VS use a 32-bit version of MS Access.
8. Make sure the current version of the MS Access database engine is installed on your computer.
9. Make sure the “Prefer 32-bit” option in project settings (**Project** menu > **Project name Properties**, where **Project name** is the name of the current project) on the Compile tab is unchecked (Figure 4.74).



**Figure 4.74: Unchecking the Prefer 32-bit setting**

### Exercise 4.8

1. List and describe elements that are contained in the connection string.
2. Weigh out the importance of testing a connection to a database before proceeding to next steps in creating the connection to the database
3. State whether it is possible to use a form in Visual Basic .NET to delete or enter data in a database. Justify your answer.
4. A database that was previously accessed from disk C was transferred to disk E to guard against security breaches. However, following that modification, nobody was able to access that database. Describe the steps you would take to address the problem.

**Project**

Develop a program, which helps people to purchase goods online. Draw objects, assign properties, attach code; use the following as a guideline to your problem:

- (a) Identify a problem
- (b) Analyse a problem
- (c) Develop an algorithm to solve the problem
- (d) Code the program using visual basic and link it to the MS Access database or spreadsheet as your data source
- (e) Test the program
- (f) Document the program

**Chapter summary**

1. This chapter has covered the major parts of Visual Basic programming, providing discussion on each key concept. Our discussion was classified into subcategories. We started on the general concept of Visual Basic .NET by defining it as a programming language that lets users create programs by manipulating program elements graphically rather than by specifying them textually. Advantages and disadvantages of Visual programming were outlined.
2. Visual Basic .NET tools, their uses, as well as the use of forms as the main part of designing VB.NET programs were discussed. Controls such as CheckBox, CommandButton, and TextBox used in VB.NET applications were described.
3. We also discussed about the project window, which is used to show files which are in the project. The properties window shows all controls a programmer can use to format the control to suit the demand of the program. Moreover, we discussed about code editor, which is used to write codes for different commands of the program designed.
4. In addition, the chapter covered on how to install the Visual Basic .NET IDE, functions of tools found in the ToolBox of the IDE, how to open IDE, saving a Visual Basic .NET program in the first time where you click save and then, save the form and write the name of the form. Also, we learned how to create our first simple program in VB.NET, “My first VB.NET program” as our first example.
5. After introducing VB.NET, the chapter discussed data types including numeric and non-numeric data types. Numeric data types involve mathematical conversions such as bytes and integers as well as non-numeric data types such as Boolean, String and Date. We covered decision control structures such as branching and iteration. Under iteration control structures, we learned the While, Do and For Each loops.
6. The chapter ended with working with data sources. In this section, we learned how to connect a VB.NET program to a database by focusing on the Microsoft Access database. Through a data source that we created, we were able to manipulate data in an Access database.

## Revision exercise

1. Develop a Visual Basic program that returns the integer part of the real value 2.67 and stores the value in variable x.
2. Write the Visual Basic programs that read two numbers using the input box and display the largest.
3. Create a Visual Basic program that reads three numbers from text boxes and displays the largest on the form label.
4. Write a VB program that displays a Day's name in English in MsgBox if the user types the number of a day in a week.
5. Write a **Do loop** program that will write numbers between 1 and 20 on the screen, and then stop. Save the project as sample4.
6. Examine the following formulas and write programs that will solve them
  - i.  $E = MC^2$
  - ii.  $F = 9/5*C + 32$
7. Write a Visual Basic program to display the following form. The word to be displayed is "Have a nice day".
8. Create a program that initially displays a grey circle (that is, without any distinctive fill color). The program will include a pull-down menu allowing the user to change the circle's fill colour to red, green or blue and clear the fill colour (thus restoring the original colour). The program will include two objects, namely a shape (the circle) and a command button, used to end the computation.
9. Design a form containing a specific title such that when we click on command1, the colour of the font will change, and when command2 is checked, the size of the font will change.
10. Design a form containing the shape and command buttons "what is this", such that when you click on the command button, the name of this shape appears in the message box.
11. Enter a text in label1 such that this text is copied to label2 after clicking on the command button "copy" and delete from label2 after clicking on the command button "delete".
12. Design a form containing a command button such that when we click on a command, form2 will appear, and form1 will disappear.
13. Design a form "at run stage" the title of form1 will be changed to "visual basic", and the back color of form1 will change too.
14. Design a form with two shapes, red and blue, and write code to replace the colour every between two shapes.
15. Design a form containing a list box and two command buttons:
  - (a) Add: Add the element to the list from the input box.
  - (b) Delete: To delete the list elements
16. Design a form with three menus and a label with a specific title. The menus contain sub-menus as follows: colour: red, blue, green, style: underline, bold, italic, size: 10, 20, 30, then write the code for each command.

# Computer security and privacy

## Introduction

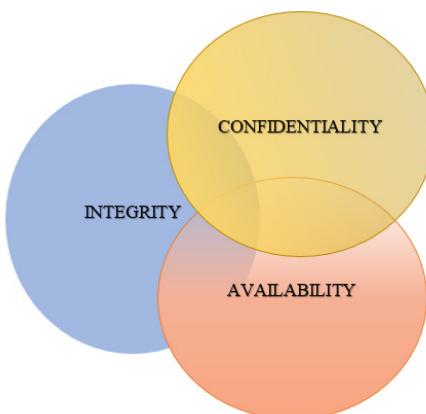
Data and information is valuable resources in digital age. If it is not well managed and protected may results into organisations functions and its operations jeopardised. In this chapter, you will learn the concept of computer security, measures taken to prevent loss and authorised data access; concepts of physical security, and troubleshooting. The competence gained will enable you to be able to used data and information ethically and responsibly.



### Think about the following:

In the modern world, an unauthorized user can access an individual's data stored in a centralized database management system without the consent of that specific individual or organization. Think of three negative impacts of the unauthorised user accessing this personal data. Suggest appropriate measures to prevent this situation.

Integrity and Availability (CIA) as shown in Figure 5.1. The organisation must ensure the security of their information systems and resources. The information system resources include hardware, software, firmware, information/data, and transmission media. The interpretations of these three aspects (CIA Triad) vary, as do the contexts in which they arise.



**Figure 5.1:** CIA triad (Confidentiality, Integrity, and Availability)

The next paragraphs present description of CIA triad.

**Confidentiality:** This ensures that unauthorized user does not get access to unauthorised computer resources. Keeping information secret arises from using computers in institutions with sensitive information such as government and industry. You can protect computer resources from unauthorised access by using an access control mechanism such as using cryptography to transform data making it incomprehensible. A cryptographic key controls access to the untransformed data, but then the cryptographic key becomes another data to be protected.

**Integrity** is the ability to ensure that the system and its data have not suffered unauthorised modification. Integrity protects the operating system, hardware and applications from being altered by unauthorised individuals in and outside the organisation. It also protects against modification, editing or changing confidential computer resources to only authorised individuals. Integrity can be defined as the trustworthiness of computer resources, usually phrased to prevent improper or unauthorized change. Therefore, adequate access controls will generally stop the break-in from the outside as well as inside and may track all activities performed by the computer users.

**Availability:** Availability refers to ensuring that systems work promptly

and services are accessible to authorised users all the time. Availability is an important aspect of reliability design because an unavailable computer resource is at least as bad as no computer resource. The aspect of availability relevant to security is that, someone may deliberately arrange to deny access to data or service by making it unavailable or unusable. For example, suppose user A has compromised the *ajira portal* or *Zanajira portal*, portals designed by the Government of Tanzania and Zanzibar, respectively, to handle new employment applications by blocking other users from submitting their curriculum vitae for the newly advertised position. If user A can supply information not generated by *these portals*, then the intended service has been compromised, and the system services are unavailable. Denial of Service (DoS) attacks can be used to block the availability of the *ajira portal*. The DoS attacks are difficult to detect because the analyst must determine if the unusual access patterns are attributable to deliberate manipulation of resources or the environment.

Furthermore, the above three security elements can be expanded to include two more parameters: authentication and authorization. Authentication ensures that computer users are validated to confirm whether the identity claim is correct. Authentication is the process or action of verifying the identity of a computer user or process. The external entity must provide information to

enable the system to confirm its identity. This information comes from one or more of the following:

1. What the entity knows (such as passwords or secret information)
2. What the entity has (such as a badge or card)
3. What the entity is (such as fingerprints or retinal characteristics)
4. Where the entity is (such as in front of a particular terminal)
5. What does the entity do (gait/movements recognitions)

Also, the authentication process consists of obtaining the information from an entity, analysing the data, and determining if it is associated with that entity. This means that the computer must store some information about the entity.

Authorisation allows us to specify whether the party should be allowed or denied access. Once we have authenticated the party in question, authorisation enables us to determine precisely what they are allowed to do. We typically implement authorization using an access controls mechanism.

### Challenges of computer security

The following are some challenges of computer security:

1. Damage or loss of computer resources
2. Requires high investment cost to achieve total computer security

3. Destruction of an organisation's reputation or branding
4. Phishing attacks may expose critical computer resources to unauthorised users
5. Software vulnerabilities could allow an attacker to gain access to computer resources.



### Activity 5.1: Read the scenario about computer security and answer the questions that follow:

Juma lives with his parents in Dodoma. Juma's father owns an Apple smartphone, generally known as iPhone, whereas Juma's mother uses an Android smartphone with the Android operating system.

1. With the help of Internet, in your own words, compare the security features available in Android and iPhone smartphones. Give at least two main security features in each of the above smartphones.
2. Write down the step-by-step procedures that you will use to ensure confidentiality, integrity, availability, authentication, and authorization for each of the above phone categories.
3. What will happen if confidentiality, integrity, availability, authentication, and authorization are missing from any of the above phone categories?



### Activity 5.2: Creating a user account and password conforming security policy

Create a user account in the windows operating system. Ensure the policy to change the password in the login is active, set the minimum password character to be 8, and set the password policy for a password to include uppercase, lowercase, special character, and numbers.



### Activity 5.3: Encrypting the folder using the VeraCrypt software tool.

#### Resources:

A VeraCrypt software tool, a user manual, a computer with Windows 10 Operating system or any software tool supporting VeraCrypt functionalities

#### Procedure:

1. Download VeraCrypt open-source tool and install it on a computer
2. Read its user manual to learn how to encrypt files and folders
3. Create files in windows
4. Put the file in the newly created folder
5. Encrypt the folder using the VeraCrypt tool by following the steps in the user manual
6. Write your observation on the encrypted folder and its contents

## Data protection

### Fundamentals of data protection

There are three fundamental components of data protection, which are discussed as follows.

**Data:** This is one of the core components of any information system. Data should be protected against unauthorized access to maintain confidentiality, integrity, availability and privacy.



### Activity 5.4: Read the scenario below and answer the questions that follow:

One day Neema got sick and decided to go to a nearby health centre for a general medical check-up. The health centre uses a health information system for patient registration, consultations, laboratory, and pharmacy. Neema provided her personal information in all stages, and the communication between the doctor and laboratory technician was automated.

Afterwards, the doctor sent the medication information to the pharmacist, and Neema was instructed by the doctor to visit the pharmacy to take her medication. Two months later, Neema found her confidential information printed on a piece of paper used for packaging bites in her street.

1. Identify confidential information that requires protection from public access or unauthorized users.
2. How will data protection policy help in protecting Neema's personal data?

**Data protection:** This is a set of methods and skills used to secure data from unauthorized access. Data protection also safeguards private and important information from compromise, corruption, and loss. The demand for data protection is also fueled by more data generated in the current digital world. The data source ranges from information systems, social media, and service automation.

**A data protection policy:** This is vital for any organization that collects, handles, or stores sensitive data. It provides guidance that prevents data loss, theft, and corruption and helps to minimize damage caused by a breach or disaster.

### Data protection policy

A data protection policy (DPP) is a security policy dedicated to standardising and controlling data use, monitoring, and management. The main goal of this policy is to protect and secure all data consumed, managed, and stored by the organization. It is not required by law, but it is commonly used to help organizations to comply with data protection standards and regulations.

Data protection policies should cover the following aspects:

1. The scope of required data protection
2. Data protection techniques and policies applied by relevant parties such as individuals, departments, devices, and the IT environment
3. Any applicable legal or compliance requirements for data protection
4. The roles and responsibilities related to data protection, including data custodians and roles specifically responsible for data protection activities

### Data privacy

Data privacy deals with properly handling data, and it is the controlling process around sharing data with a third party so that only authorized users can access it. Data privacy is a set of rules that govern how data is acquired and managed based on its sensitivity and value. Data privacy is applied to Personal Health Information (PHI) and Personally Identifiable Information (PII). It includes financial data, medical records, social security or ID numbers, names, birth dates, and contact information.

Concerns over data privacy apply to all sensitive information that firms handle, including customers, shareholders, and employees. This information is frequently critical in corporate operations, development, and finances. Data privacy helps to ensure that sensitive information is only available to authorised users. It secures data from being used by criminals maliciously and assists enterprises in meeting regulatory requirements.

### Computer privacy

Computer privacy is just the safeguard of a computer whereby it is controlled from being used by other persons who are not authorised. Privacy should be maintained when using computers and information systems. The use of an information system should be kept safe and secret, and no one should be allowed to engage in illegal activity while accessing that information.

## Data protection and privacy regulation

### International and national data protection regulation

Data protection and privacy regulations vary between countries, states, and industries. For example, China created a data privacy law that went into effect on June 1, 2017, and the European Union's (EU) General Data Protection Regulation (GDPR) went into effect in 2018. China's data privacy law was designed to protect the personal data of individuals and increase data security. It specifies the scope of personal information; clarifies the legal bases for processing personal information; lays down the obligations and responsibilities imposed on processors, and imposes stringent requirements on data localization, safeguarding China's interests in the case of cross-border transfer of personal information. Apart from the Personal Information Protection Law, China also has other laws such as Data Security Law and Cyber Security Law, to ensure cyber security and data protection.

On the other hand, GDPR was designed for the protection of natural persons related to processing the personal data in the European Union. Also, it enhances how people access their information and limits how organisations and companies must use personal data in an integrity friendly way.

Tanzania has enacted the Electronic and Postal Communications Act of 2010 (EPOCA), which provides security and protection guidance for Tanzania's electronic and postal communications

services. EPOCA was established to make provisions for the enactment of electronic and postal communications law to keep abreast with developments in the electronic communications industry. Tanzania also has the Electronic Transactions Act of 2015 (ETA) and the Cybercrime Act. The ETA provides the legal recognition of electronic transactions, e-Government services and the use of Information and Communication Technologies in the collection of evidence; whereas the Cybercrime Act of 2015 makes provisions for criminalizing offences related to computer systems and Information Communication Technologies to facilitate the investigation, collection, and use of electronic evidence in Tanzania. Compliance with one set of regulations does not guarantee compliance with all laws. Additionally, each directive contains numerous clauses that may apply to one case but not another, and all rules are subject to change. This level of complexity makes it challenging to implement compliance consistently and appropriately.

The application of these laws criminalizes and penalizes several cyber activities such as data espionage, publication of child and adult pornography. Others include publication of pornography, publication of false, deceptive, and misleading information, production and dissemination of racist and xenophobic material. In addition, initiating transmission of or re-transmission of unsolicited messages and violation of intellectual property rights and other types of cybercrimes constitute

activities that can be criminalised and penalised. The cybercrimes Act 2015, came after significant impacts such as financial loss, fraud and cyberbullying to the public and other stakeholders. Non-compliance may result in reputation damages and monetary fines, depending on the violation as instructed by each law and governing entity.



### Activity 5.5: Searching for cybercrime Acts and related regulations

#### Resources:

Library resources, internet connectivity, the Tanzania cybercrime act and other related cyber-related regulations.

#### Procedures:

1. Visit a library or search the current Tanzania cybercrime act on the Internet.
2. Study various laws that govern information security.
3. Write one section and its subsections of the act stating its phrase and explain how it addresses the cyber-related activities/ offences in the society

### Importance of data protection

Data protection has the following importance:

- (a) The essential values of data protection are safeguarding and making available data under all circumstances. That describes the operational backup of data and Business Continuity Disaster Recovery (BCDR).

- (b) ICAT protects organizational assets from fraud cases such as hacking, phishing, and identity theft.
- (c) It gives consumers more confidence to buy items or things online. As a result, it increases revenue for enterprises both offline and online.

### Data protection technologies

There are various technologies and practices to protect data in an organisation. The most commonly used methods and technologies are as follows:

*Data discovery:* This entails determining which data sets exist in the organisation, which are essential to the organisation role, and which contain sensitive information that may be subject to compliance regulations.

*Data Loss Prevention (DLP):* A set of strategies and tools that you can use to prevent data from being stolen, lost, or accidentally deleted. Data loss prevention solutions often include several mechanisms to protect against and recover from data loss. Most modern storage equipment with built-in data protection provides built-in disk clustering and redundancy.

*Encryption:* Changes the data content into non-meaningful texts and uses an algorithm that can restore the original content using the correct encryption key. It protects information from being accessed by rendering it impossible to read even if the data is stolen. More information can be found in the data encryption guide.

*Storage with built-in data protection:* Modern storage equipment have disc clustering and redundancy to ensure data recovery during a disaster.

*Screenshots to backups and snapshots:* These are complete images of a protected system, including data and system files. The image can restore an entire system to a specific time in the past.

### Relationship between information security, cybersecurity, system security, and network security

These terms sometimes are used interchangeably, but it depends on the area to be the emphasis for protection. Let us look at the meaning and relationship between these terms:

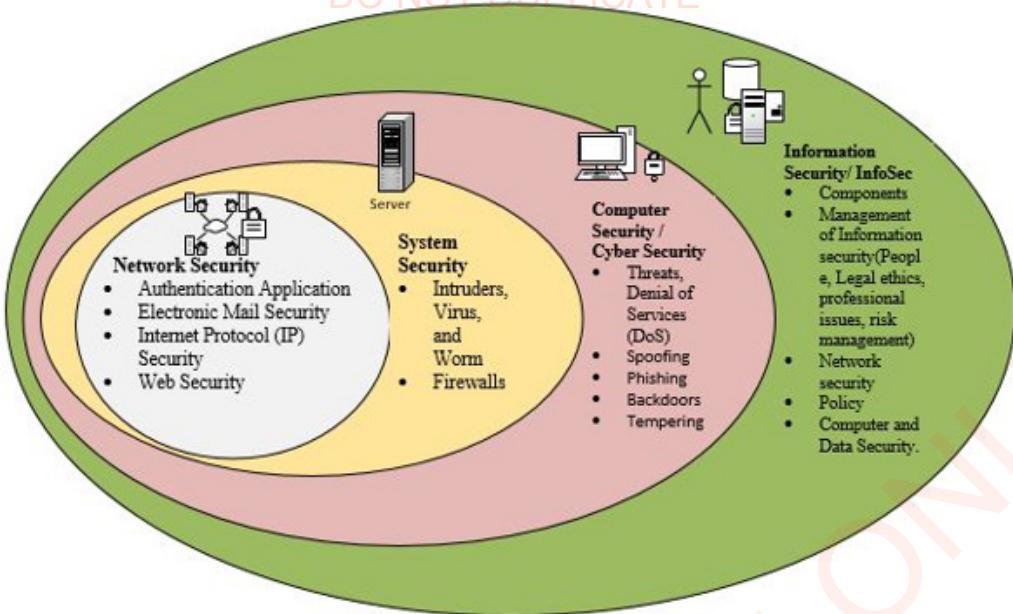
*Information security:* Information security (also known as InfoSec) ensures that both physical and digital data are protected from unauthorized access, use, disclosure, disruption, modification, inspection, recording, or destruction. Information security differs from cybersecurity in that, InfoSec aims to keep data safe in any form, not only data in digital format. For example, suppose you have a paper document. In that case, securing this document is an information security responsibility. Still, if you scan this document and transfer it into a digital file, this digital file's security is a cybersecurity responsibility.

*Cybersecurity:* It is all about protecting data that are found in electronic form

(such as computers, servers, networks, and mobile devices.) from being compromised or attacked. Cybersecurity involves identifying the critical data, where it resides, its risk exposure, and the technology you must implement to protect it.

Cybersecurity, as a subset of information security, is the practice of defending your organization's cloud, networks, computers, and data from unauthorized digital access, attacks, or damage by implementing various defence processes, technologies, and practices. With the countless sophisticated threat actors targeting all types of organizations, your IT infrastructure must be secured to prevent a full-scale attack on your clouds, networks, or endpoints and risk exposing your company to fines, data losses, and damage to reputation.

*Network security:* It is a subset of cybersecurity which consists of the policies, processes and practices adopted to prevent, detect and monitor unauthorized access, misuse, modification, or denial of a computer network and network-accessible resources. Network security involves securing data access in a network, which the network administrator controls. Figure 5.2 shows the difference between information security, computer/cyber security, system security and network security.



**Figure 5.2:** Relationship between network security, system security, computer security/cybersecurity and information security

### Identification, authentication, accountability and non-repudiation

#### Identification

Identification is the process that identifies uniquely a user of a system. Identification is merely a claim of identification and does not indicate that the claim is correct or that any benefits associated with the identity are awarded if proven accurate.

The typical examples are that you can bring out your identity cards from jobs or schools, a driver's license card in the Tanzania vehicle, a motor vehicle insurance card, a national ID card, birth certificates, academic certificates, and an NHIF card for health insurance. Such cards are routinely used to prove our identity, whether genuine or accurate, when making purchases, dealing with government officials like Traffic Police

officers and offices like hospitals, registering for treatment, registering for school, and performing various other tasks. In many cases, identification cards verify our identity while doing these things.

#### Authentication

Authentication is the procedure for determining whether or not a claim of identity is genuine. It is vital to remember that authentication and verification are not the exact words and that verification is a much weaker security test than authentication. Identity verification is a step beyond identification, but it is still a step short of authentication. For example, you are asked to show a driver's license, NHIF card, birth certificate and National identity card, typically used for identity verification rather than authentication. You can use a variety of aspects to

perform authentication. Authentication is an essential part of something you have, something you do, and where you are. When you are prompted to enter the PIN from the ATM card associated with the card, you complete the transaction's authentication portion. These are some of the identification and authentication methods used in daily life.

### **Accountability**

Accountability refers to the ability of a system to deliver audit trails of every transaction. The system managers must be able to offer information on every transaction that has taken place because they are subject to external scrutiny. Hence, the action impacting security may be traced back to the accountable party. Audit trails must be stored in selective manner and protected to ensure their integrity.

### **Non-repudiation**

Non-repudiation is the ability to ensure that computer/network users or participants do not deny (repudiate) their actions. Hence, assurance of security service that provides proof of service, origin and delivery is done by non-repudiation. In real life, the sender may deny ownership of the exchanged digital data that originated with them. This service ensures that digital data cannot be repudiated by providing difficult-to-denry proof of origin via digital signature and encryption algorithms.

### **Models of information security**

*A security model:* It is a framework that determines how security is implemented by implementing security policies. Without efficient and robust security models, no organisation can secure sensitive information or data.

**DRAFT** *The model's principal goal:* Provides the necessary understanding for the successful and effective implementation of essential protection requirements.

*Information security models:* These are procedures used to validate security policies. They are designed to provide specific instructions for a computer to perform a security program's critical security processes, procedures, and concepts. These models might be either intuitive or abstract. Security models, for example, define the road map for operating system security.

These security models are vital components that must be considered when engineering security systems and policies in your organization. They should define the access rules necessary to instantiate the said policy and define the entities (subjects governed by an organisation's security policy).

*Five security models:* Define the rules and policies that govern integrity, confidentiality, and data protection. Confidentiality through information integrity and access controls is the main focus and reason for the security model implementation.

The following are examples of security models:

- (a) Bell-LaPadula
- (b) Biba
- (c) Clark-Wilson
- (d) Chinese Wall model
- (e) Harrison-Ruzzo-Ullman.

### **Bell-LaPadula (BLP)**

The Bell-LaPadula (BLP) security model is also, called the multi-level model, was proposed by Bell and LaPadula for

enforcing access control for computer systems. In this model, each subject (user) and each object (resources such as, files) is assigned one of the security levels: top secret, secret, confidential, classified and unclassified. Therefore, the unclassified user can not access data at the upper levels (classified, confidential, secret and top secret), and the files tagged as top secret can not be accessed by lower levels (secret, confidential, classified and unclassified). This kind of access control is also called mandatory access control, which restricts access to an object based on the sensitivity of the information contained in that object and formally authorises the subject to access information based on its sensitivity.

The Bell-LaPadula model has two limitations:

1. It provides confidentiality only (no integrity and authentication).
2. It provides no methods for the management of classification (it assumes all data are assigned with a classification and will never change).

### **Biba model**

The Biba security model was published to handle integrity issues which BLP Model did not address. The goal of the Biba model was to prevent unauthorised users from making modifications to particular computer resources.

### **Clark-Wilson**

The Clark-Wilson model distinguishes between two types of objects, namely Constrained Data Items (CDIs) and Unconstrained Data Items ((UDIs)). It also addresses two types of transactions,

which are Integrity Verification Procedures (IVPs) and transaction procedures (TPs). The IVP's job is to ensure that the TPs produced by CDIs are in a valid state, and all TPs must be certified to make a proper transformation. Authorized TPs can only manipulate CDIs. The Clark-Wilson model enforces integrity goals by using access triple (subjects, software transaction procedure and object), separation of duties, and auditing. It uses well-formed transactions (through access triple) and separation of user duties.

### **Chinese Wall model**

The Chinese Wall model's theory is based on a conflict of interest, which states that a single user should not have access to confidential information belonging to two separate interested and participating stakeholders. User behaviour influences access control policies. In other words, once one side's data is accessed, the other side's data becomes inaccessible or unavailable.

### **Harrison–Ruzzo–Ullman (HRU)**

The BLP model lacks mechanisms for changing access rights and deleting subjects and objects due to the results. The HRU model addresses these issues by defining an authorization system to assign access rights and verify compliance with any policy, preventing unauthorized access.

### **Attacks**

An attack is an information security threat involving the unauthorized acquisition, modification, destruction, removal, implant, or disclosure of information.

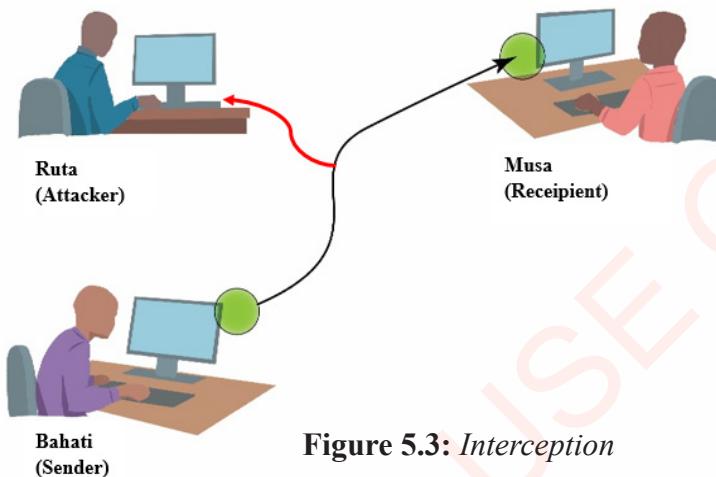
## Types of attack

You can categorise the different forms of attacks and group them into four categories: interception, interruption, modification (alteration), and fabrication.

### Interception

Interception is the act that occurs when an individual who is not authorised gets access to files, data, conversation, e-mail,

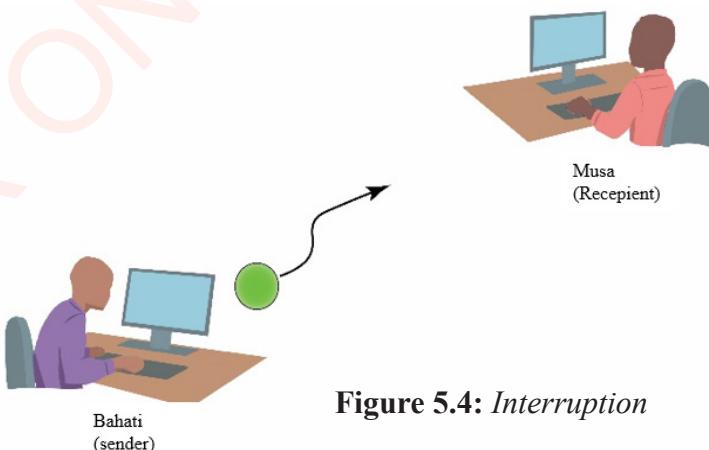
etc., between two individuals who are communicating. It can be performed either when data are in motion or stationary. In the interception, the attack occurs before the information reaches the intended recipient. Interception attacks, when executed correctly, can always be challenging to detect. It is an attack on confidentiality. Figure 5.3 shows the interception attack.



**Figure 5.3: Interception**

### Interruption

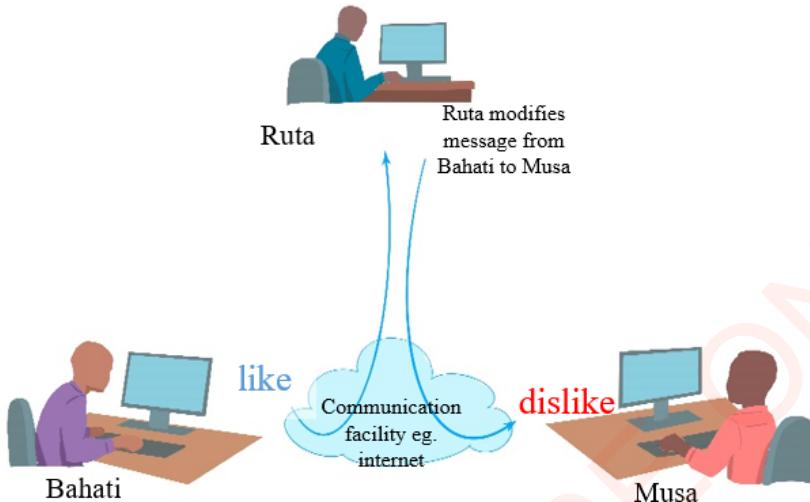
This occurs when the individual who is not authorised to access data gains access to information between two communicating individuals making the services unavailable temporarily or permanently. The person interrupts the information such as, files, conversations, E-mails, etc., by blocking communication between two individuals, making the services unavailable, as shown in Figure 5.4. Interruption is performed when the data are in either motion or stationary. It is an attack on availability.



**Figure 5.4: Interruption**

**Modification**

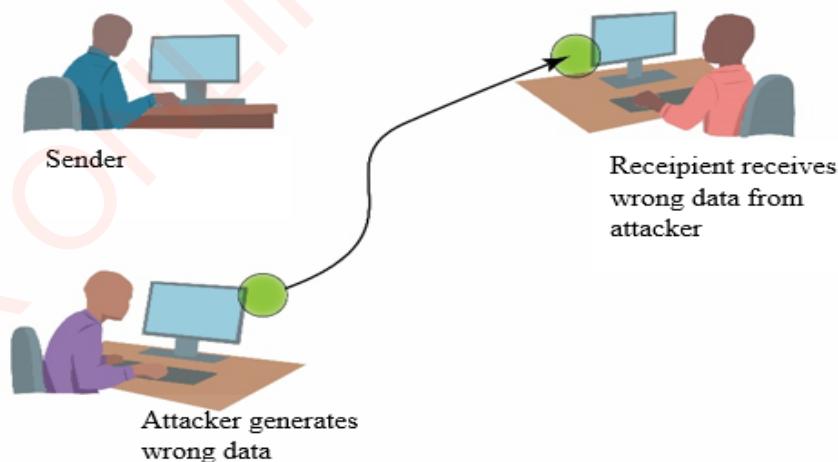
This is editing the content of information from an individual without the person or organisation's consent. This is referred to as an integrity attack because the data integrity is damaged. When the data is compromised, this can cause a severe problem for an individual or organisation. Figure 5.5 shows the modification attack.



**Figure 5.5: Modification**

**Fabrication**

Fabrication attack refers to the act of generating incorrect data and insert it alongside authentic data. This involves generating data, processes, communications, or other similar activities within a system. Fabrication attacks initially affect integrity but can also be considered availability attacks. It is also an attack on authenticity. The fabrication attack is illustrated in Figure 5.6.



**Figure 5.6: Fabrication**

## Threats

A threat is defined as something that has the potential to harm a computer or create negative consequences. Threats in cybersecurity are commonly specialized to specific programs or operating systems. A virus, for example, may pose a risk to a Windows operating system, but it is unlikely to have any effect on a Linux operating system.

### Man-made threats

These are threats that can harm computer software and hardware; resulting in a malfunction of the computer system. Man-made threats involve the following:

#### Malware

Malware is a term used to describe malicious software, sometimes called malicious code.

Malware intends to disrupt, damage, or gain unauthorized access to a computer system. It is purposefully designed to harm, disrupt, steal, or do something “wrong” or illegal with data, hosts, or networks. Examples of malware include viruses, worms, trojan horses, logic bombs, and spyware.

Malware comes in various aspects, each with its method of infecting and spreading in computers. Malware can infect computers by being packaged with other programs or attached to files such as macros.

#### Types of malware (malicious code)

The common types of malware are viruses, worms, Trojans, Bots, Logic bombs, Rootkit, spyware, and adware.

Some descriptions of types of malware are as follows:

A virus is a piece of code capable of replicating itself and typically has a negative impact, such as distorting the system or destroying data. It spreads by inserting multiple copies of its own into the programs or data files that it infects. User input usually activates viruses, such as opening a file or running software.

Viruses are divided into two categories, which are compiled viruses and uncompiled scripting viruses.

For example, an operating system executes a compiled virus. File infector viruses attach themselves to executable programs such as, boot sector viruses, which infect the master boot records of hard drives or the boot sectors of removable media and multipartite viruses, which combine the characteristics of file infector and boot sector viruses.

Scripting viruses infect scripts that are understood by scripting languages processed by Operating System services, while macro viruses abuse the capabilities of applications’ macro programming language to infect application documents and document formats.

#### Worms

A worm is a self-replicating identity program that runs without the user’s knowledge. Worms are categorised into worms that infect networks and worms that infect bulk mail. A network service worm spreads by infecting other systems by exploiting the vulnerability of a network service.

### **Trojan horse**

A Trojan horse is a self-contained, non-replicating program that serves a malicious intent despite its innocent appearance. It masquerades as an authentic software or application to capture user information such as username, password, biometric information and passcodes. The information captured by hackers can be used to perform other malicious activities.

### **Logic bombs**

Logic bombs are malicious programs typically installed on a design by an authorized user. They are composed of some code or a program that remains inactive for an extended time, waiting for a logical condition to be met before executing its malicious payload.

### **Spyware**

Spyware is a sneaky program that attempts to steal your information such as credit card, banking information, web browsing data, and password to various accounts. While spyware is not intended to cause harm, it can harm your computer's performance over time. Spyware is typically bundled with free software and installs itself alongside the program you intend to use. Sudden changes in your web browser such as, redirections of your search attempts and the frequent display of pop-ups, are among the signs of spyware.

### **Rootkit**

A rootkit can be a single program or a group of programs designed to take complete control of a system. Hackers use this type of malware to gain access to all of the capabilities of a system

administrator from a remote location. Rootkits are extremely sophisticated because they make hackers extremely difficult to locate. They are frequently used to penetrate and imprison other computers as zombies, forcing them to attack other machines, distribute spam, or steal passwords. When attempting to track down the creator of a rootkit, the search usually ends with the first computer used as a zombie while the hacker remains undetected.

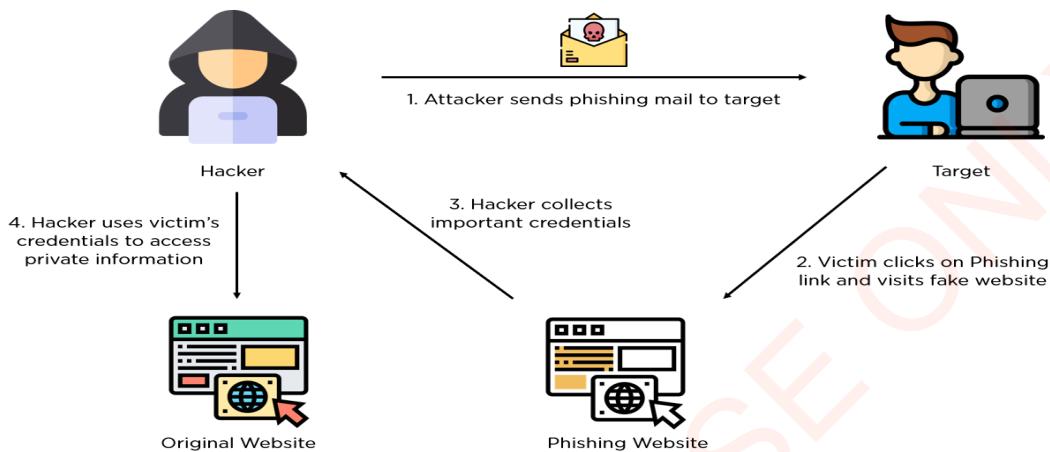
### **Bots**

Bots, also known as Botnets, are automated processes interacting with other network services. They are derived from the word "robot." Bots frequently automate tasks and provide information or services that humans would otherwise perform. Bots are typically used to gather information (such as web crawlers) or to interact with instant messaging (IM), Internet Relay Chat (IRC), or other web interfaces automatically. It can also be used to interact with websites dynamically. Bots can be used for either good or evil purposes. A malicious bot is a self-propagating malware that infects a host and connects to a central server that serves as a Command and Control (C&C) centre for an entire network of compromised devices, or "botnet."

### **Phishing attack**

Phishing attacks use fake communication, such as an email, to trick the receiver into opening it and carrying out the instructions inside, such as providing a credit card number. "The goal is to steal sensitive data like credit card and login information or install malware on the victim's machine."

The term “phishing” comes from the analogy of early internet criminals using lures to “fish” for passwords and financial information from a large sea of unsuspecting internet users. The prefix “ph” in this terminology has gradually faded. Phishing refers to attempting to contact a victim via email or phone while impersonating a legitimate institution to trick the individual into providing sensitive information such as banking information, credit card details, and passwords. Nevertheless, the individual’s data can then be used to access their account, resulting in identity theft and financial loss. Figure 5.7 shows a phishing attack.



**Figure 5.7: Phishing attack**

### Man in the Middle Attack

Man in the Middle Attack (MITM) attacks when a visitor uses an unsecured public network. Here, attackers insert themselves between the visitor and the web, and then use malware to install software and use data maliciously.

A man in the middle attack can be carried out after a hacker performs Address Resolution Protocol (ARP) spoofing, which is done by sending false ARP messages over the managed network to penetrate the local area network. When done correctly, the forged ARP

messages allow the hacker’s MAC address to be successfully linked to an IP address of a genuine user or an entire server in a targeted network. Once the hackers link with MAC address to a legitimate IP address, you will be able to receive all data which other users send to the IP address. When the user/victim communicates with another user, the attacker inserts himself in the middle of the conversation by sniffing packets and producing MITM. This action involves utilities used to attempt MITM attacks, such as SSL Strip, burp suite, and browser exploitation framework.

Figure 5.8 illustrates man in the middle attack.

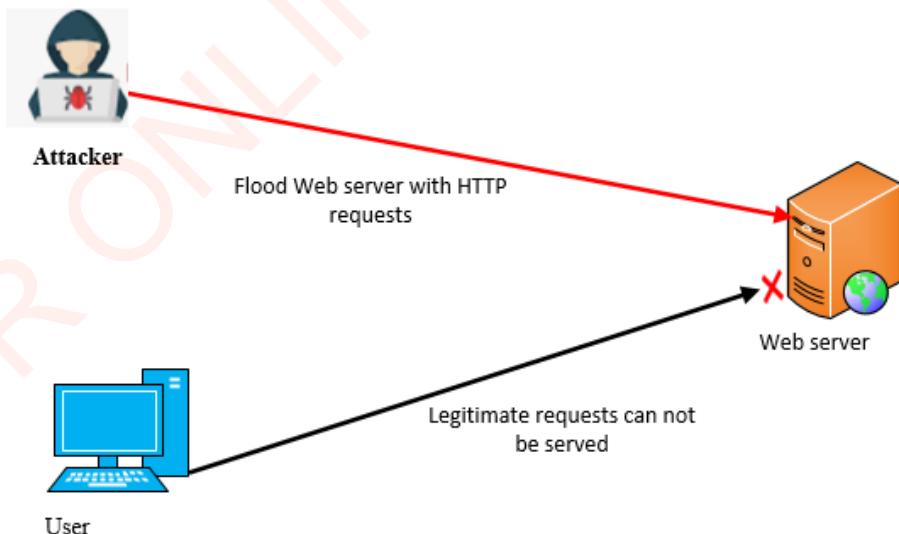
**Figure 5.8: Man in the Middle attack**

### Denial of Service attack

Denial of Service (DoS) is an attempt by attackers to deny service to the intended purposes of the service. It occurs when a computer resource is overburdened with more requests than it can handle, consuming all available bandwidth and resulting in server overload, as illustrated in Figure 5.9.

DDoS is a malicious attempt to disrupt the normal operations of targeted computer systems or services by overwhelming the target or its surrounding infrastructure with a flood of requests. The affected computers will not be able to communicate as intended.

The most popular DDoS attacks are the Ping of Death, smurfing, teardrop, and the land. DDoS attacks can be local, causing LAN machines to fail. They can initially come hundreds of miles away on the Internet, equating to trying to penetrate electronic attacks.

**Figure 5.9: Denial of Service**

## SQL Injection attack

SQL Injection is a type of cyber-attack in which a hacker manipulates a database with SQL (Structured Query Language) code to access sensitive critical data. SQL statements are injected as input strings. SQL injection is a severe and vulnerable cyberattack that can lead to a high level of compromise, typically allowing the running of any database query. It is a web-based application attack that connects to database backends and sidesteps firewalls. Consider the SQL Injection attack, for example, a login screen with two input fields (Login and Password) whose values are supplied by the user. When a user clicks the Login button, a request to access the website is sent to the server and the provided credentials. Another example is the HTML code display for the HTML from the server-side Login and Password input fields are accessed here via the '\$fLogin' and '\$fPassword' variables, respectively.

This is a simple example of gaining access with a username and a password. In this example database has a table named *users* with the following columns: *id*, *username* and *password*.

```
# Define POST variables
```

**Username1 = request.**

**POST[‘username’]**

**Password1 = request.**

**POST[‘password’]**

```
# SQL query vulnerable to SQLi
```

```
sql = “SELECT id FROM users  
WHERE username=”” + username1 +  
“” AND password=”” + password1 + “””
```

**# Execute the SQL statement**

**database.execute(sql)**

These input fields are vulnerable to SQL Injection. For example, attackers can use a trick involving a single quote and set the **password1** field to:

**password’ OR ‘1=‘1**

As a result, the database server runs the following SQL query:

```
SELECT id FROM users WHERE  
username=’username1’ AND  
password=’password1’ OR ‘1=‘1
```

Because the OR 1=1 is always evaluated to the TRUE statement, the entire **WHERE** clause will be TRUE and returns the first id from the user’s table, regardless of username and password. The first user id in a database is very often the administrator. In this way, the attacker bypasses the authentication and gains administrator privileges. They can also comment out the rest of the SQL statement to control the execution of the SQL query further (See Figure 5.10).

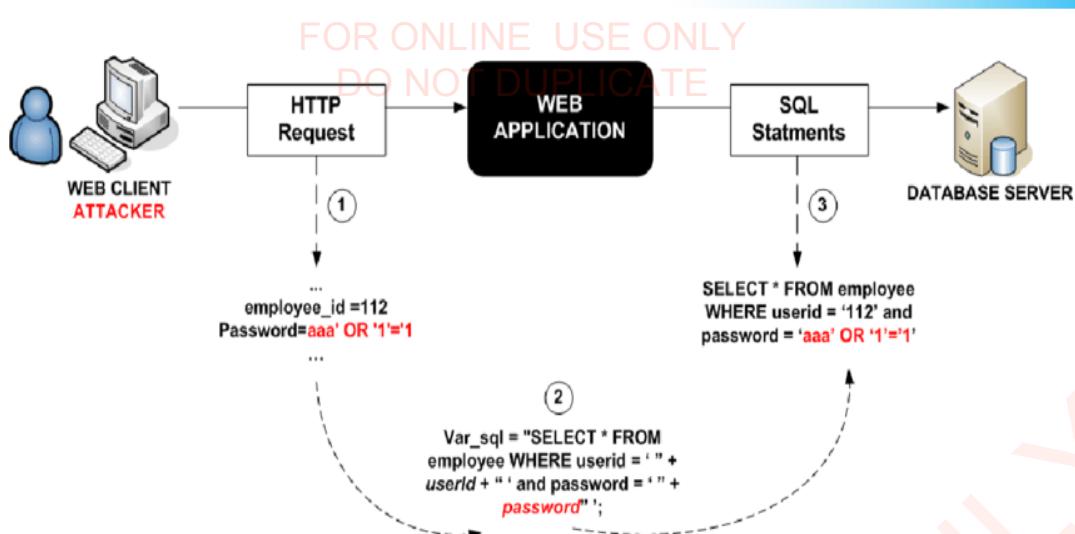


Figure 5.10: SQL injection attack

### Exercise 5.1.

Answer the following question

1. Briefly explain five security models.
2. Describe four types of attack payload.
3. Show steps by steps how to conduct SQL injection.
4. Discuss prevention measures for SQL Injection Attacks.
5. Apart from username and password, which variables are candidates for SQL Injection?
6. I am using client-side JavaScript code for checking user input. Is that not enough? Why?

### Zero-day exploit

A zero-day attack, also known as Day Zero, is an attack that takes advantage of a severe possible software security vulnerability that the vendor or developer is unaware. To limit the threat to software users, the software developer must act quickly to address the fault. One example is the Microsoft Exchange attack that compromised hundreds of thousands of organizations. Figure 5.11 illustrates a zero-day exploit attack.

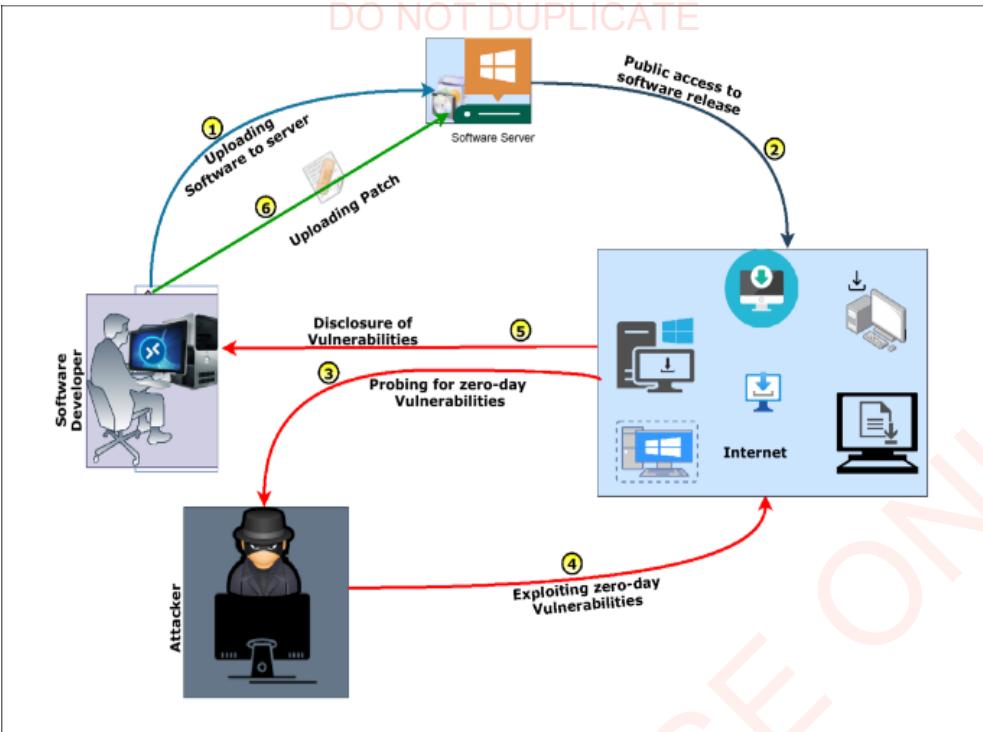


Figure 5.11: Zero-day exploit Attack (Kumar, V. & Sinha, D (2021)

### Ransomware

Ransomware is malware created to stop a user or organization from downloading or accessing data on their computers. Once installed on your computer, it encrypts all of your files and holds your data hostage until you pay the ransom demanded by the hacker. Ransomware spreads viruses, and even if you pay the ransom, there is no guarantee that your data will be recovered.

### DNS Attack

The Domain Name System (DNS) attack redirects someone to a malicious website where a hacker can quickly harvest private information or user credentials using a website's IP address. Correspondingly, DNS is in charge of converting domain names (such as www.morogorotc.ac.tz) that you are familiar

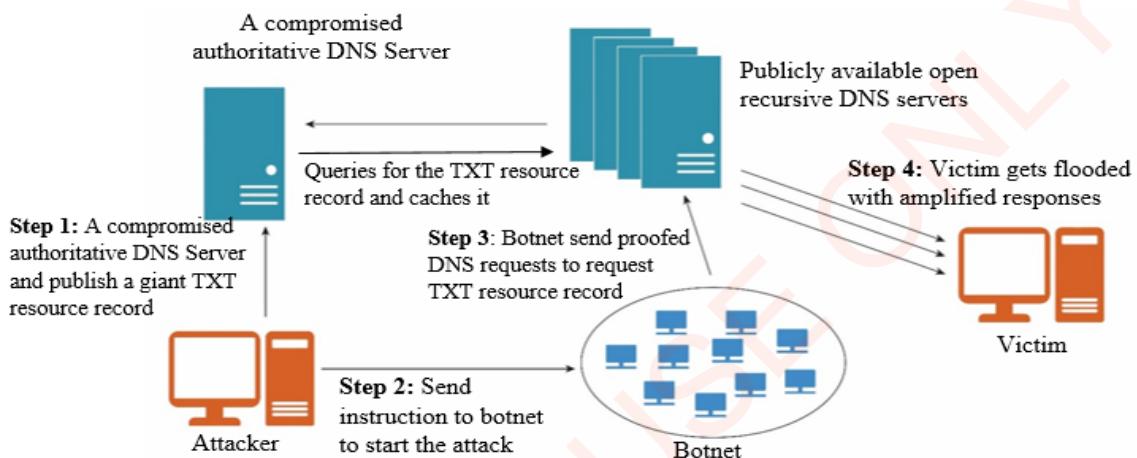
with and IP addresses that computers and routers possess. DNS poisoning is one of several techniques attackers use to redirect traffic to an individual website that can be used later to steal personal information. Here the attacker or hacker needs to be on the same network as the target. To acquire access to that network, a hacker can simply search for a weak password on a connected machine connected to that network, which can even be done remotely. An example of a DNS attack is shown in Figure 5.12.

For example, the scenario whereby an attacker force executing a DNS poisoning attack:

First, the attacker creates a phishing website. It spoofs a bank that we will call Mwananchi Bank. The attackers want to lure users there, so they can steal their usernames and passwords and use

them on the actual bank website. In this case, attackers create a DNS server and put two false records in the DNS cache. The first is for the Mwananchi Bank website, pointing to a fake site rather than the accurate bank site.

The hacker's DNS server then offers to perform a zone transfer, exchanging all data with the target server. This information includes the forged Mwananchi Bank address. The target DNS server has an entry for Mwananchi Bank. Instead of the actual Mwananchi Bank website, the target DNS server has access to Mwananchi Bank, pointing to the hacker's website. If users on that Network enter the URL for Mwananchi Bank, their DNS server will redirect people to the hacker's site.



**Figure 5.12: DNS Attack**

## Natural threats

All threats that occur naturally without human intention, such as earthquakes, fire, cyclones, and floods, pose additional threats to IT systems, data, and infrastructure. Building and computer hardware damage can result in the loss or corruption of customer records/transactions.



### Activity 5.6: Analysing security features of Antivirus

1. Use the Internet to download any four (4) free versions of Antiviruses.
2. Install them on a computer running the Windows Operating system.
3. After installation, run them one by one and take note of their basic protection features. When you finish, try to compare their security features and suggest the best antivirus depending on your research.

### **Methods to protect computer systems against viruses/worms/Trojan horse**

- Scanning critical host components such as start-up files and boot records.
- Real-time monitoring of host activities to detect unusual behaviour; for instance, scanning all email attachments for known malware.
- Antivirus software should monitor activity involving the applications most likely to be used to infect hosts or spread malware.
- Examining files for malicious programs. Depending on the organization's security needs, antivirus software on hosts should be configured to scan all hard drives regularly to ensure no file system infections.

Steps to protect computer systems against human intervention are as follows:

- Encrypt data
- Install antivirus and antispyware programs
- Install firewall
- Back up data periodically
- Keep the computer in a safe environment (with good physical security)
- Install security cameras like CCTV in your computer laboratory

### **Risks management**

#### **Risk**

Risk is defined as the chance of something terrible happening. For the risk to happen in a given environment, there must be both a danger and a vulnerability threat

that can exploit. In addition to that, the risk is the possibility that something will go wrong or an unfavourable event will occur to affect the availability, confidentiality, or integrity of business or personnel information. For example, if you plan to start a fire in a wooden structure, you have a danger (the fire) as well as a weakness (the vulnerability) (the wood structure). There is a high risk here that the fire will burn the wood structure. Similarly, if you face the same fire threat but our building is made of concrete, our threat seems to be no longer correct because it lacks weakness to exploit. You can say that a sufficiently hot flame could destroy the concrete.

#### **Risk management**

Risk management is an assessment that defines the risks in your organization, future technologies, and methods to ensure that controls are in place to protect against not only cyber threats but also, natural and human-made disasters. Moreover, to compensate for risks in our surroundings, a risk management process must be implemented and followed. This program must be managed by the organization's senior leaders and put in place by everyone.

#### **Computer risk**

A computer risk is anything else that has the potential to harm a computer user's data. The worth of this information varies depending on the computer system. A hacker can steal data through various methods such as attacking the Operating System by using Trojan horses or other malware software.

## Types of computer risks

You can categorize computer risks as the risks that happen to hardware, software, network, and human errors.

### **Computer risks in hardware**

Computer risks in hardware result from unsafe computing habits such as the improper shutdown of the computer or damaging hardware part that generates threats to the computer itself and misconfiguration.

### **Computer risk in software**

Computer risk in software causes damage to the computer internally (operative system damage) without the permission or knowledge of the user, like Viruses, worms, ransomware, spyware, and Trojan horses. Hence, those activities can be done online by human error or offline by disgruntled employees.

### **Computer risk in Network**

Computer risk in networks is anything that may damage or steal data from your computer or allow someone else to access your computer without your consent. This includes online hacking activities, ransomware, rogue security software, the man-in-the-middle attack, data leakage and denial-of-service attack.

### **Risk assessment process**

The risk assessment process refers to assessing, investigating, setting priorities, and developing plans to prevent threats to an organisation's assets and earnings.

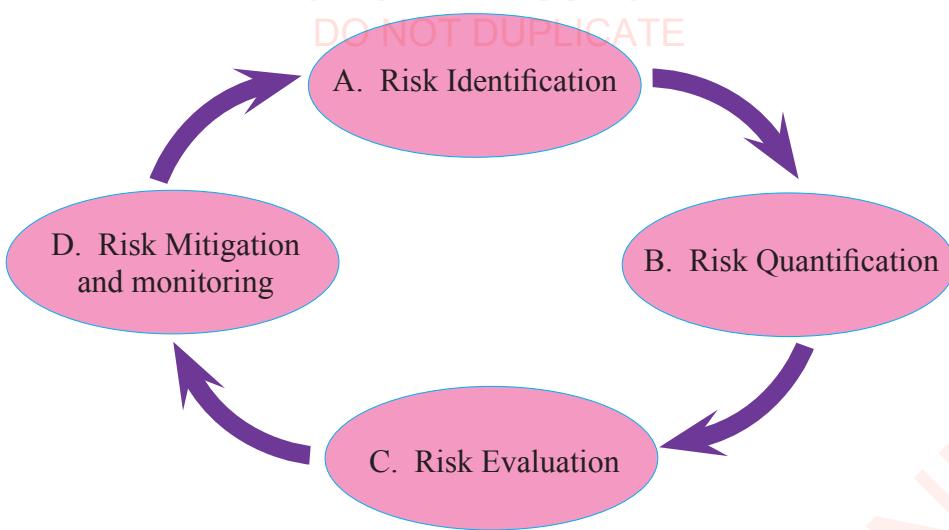
It examines, discovers, and identifies system and applications security

measures and weaknesses. During this process, systems and applications are reviewed for security measures to determine the effectiveness of deployed security layer to withstand attacks and misuses. Assessing the vulnerability helps to identify the weaknesses and flaws of vulnerabilities that could be exploited; and therefore, develop the countermeasures strategies against or provide an additional security layer.

The following procedures should be followed in risk assessment process:

- (a) Identify threats and risks for a specific asset. The overall assessment results from a threat and a vulnerability combined. Vulnerability in the absence of a corresponding risk, or a danger in the lack of a related vulnerability, is not a risk. For example, you may suggest items as both a potential threat and a vulnerability.
- (b) Quantifications which involves evaluating the identified risks and determine their likelihood and severity. This also entails validation, measurement and the analysis of the risks using mathematical techniques
- (c) Build a risk profile for each assess and evaluate the risk.
- (d) Create a mitigation plan with security controls to eliminate or mitigate the impact of each risk, monitor risks, threats, and vulnerabilities on an ongoing basis.

Figure 5.13 shows the procedures that should be followed in the risk assessment process.

**Figure 5.13:** Risks assessment process

### Vulnerability

A vulnerability is a flaw that cybercriminals can exploit to gain unauthorized access to a computer system. On the other hand, vulnerabilities are flaws that can be exploited to cause harm. In essence, they are security flaws that hackers could use to harm us. A vulnerability can be a flaw in the operating system or application you are using, or it can be a physical flaw in our office building, a data centre with an estimated population that exceeds the capacity of its air conditioning system, or a lack of emergency power.

### Vulnerability analysis

Vulnerability analysis entails identifying weaknesses in an environment, design flaws, and other security concerns that may lead to the misuse of an operating system, application, or website, which may lead to exposure to the risks of being attacked.

### Vulnerability assessment

A vulnerability assessment involves systematically examining an

information system's security flaws. It includes discovering weaknesses in an environment, design flaws, and other security concerns that can cause an Operating System, application, or website to be misused. The vulnerabilities are misconfigurations, default configurations, buffer overflows, Operating System flaws and open services. Network administrators can scan a network for vulnerabilities using various tools such as web application scanners and Nmap tools, which are open source and free to use.

### Types of vulnerability assessments

*Active assessments:* It is the process of vulnerability assessment which includes actively sending requests to the live network and examining the responses.

*Passive assessments:* These include packet sniffing to discover vulnerabilities, running services, open ports, and other information.

*External Assessment:* It assesses hacking's viewpoint to find vulnerabilities and exploit them from outside.

Authorization and access control  
techniques

### Incident response

If our risk management efforts fail, incident response is available to respond to such events. Incident response should focus primarily on the items you believe will cause us pain as an organization, which we should now be aware of based on our research. Risk-management initiatives and the possible reactions to such incidents should be based on documented incident response plans reviewed regularly, tested, and practised by those expected to put them into action in the field. The incident response process, at a high level, consists of the following phases:

- (a) Preparation
- (b) Detection and analysis
- (c) Containment
- (d) Eradication
- (e) Recovery
- (f) Post-incident activity

#### **Preparation**

The incident response preparation phase includes all of the activities you can perform before the incident to prepare us better to handle it. Typically it entails putting incident response and operating policies and procedures, conducting training and education for incident handlers and those expected to report incidents, performing incident response exercises, developing and maintaining documentation, and other activities.

#### **Detection and analysis**

In this phase, detection is done as the result of frequently monitoring or

alerting operations based on the output of a security tool or service deployed. These tools include Intrusion Detection System (IDS), Anti-Virus, firewall logs, proxy logs, and alerting Security Information and Event Monitoring (SIEM).

#### **Containment, eradication and recovery**

Most of the work to solve the incident occurs during the containment, eradication, and recovery phase, at least in the short term.

**Containment:** This entails ensuring that the situation does not cause any more damage than it already has or, at the very least, that any ongoing harm is minimized. Suppose the problem is a malware-infected server that a remote attacker is actively controlling. In that case, this may entail disconnecting the server from the network, putting firewall rules in place to block the attacker, and updating signatures or regulations on an Intrusion Prevention System (IPS) to stop the malware traffic.

**Eradication:** During eradication, you will attempt to eliminate the issues affecting our environment. You have already isolated our malware-infected server and disconnected it from its command-and-control network. Users must remove the malware from the server and ensure it is not visible anywhere else in our environment.

**Recovery:** Lastly, you must come back to a stable state as you were before the incident (before the issue started), if you did not detect the problem immediately. The recovery includes restoring devices or data from backup media, rebuilding systems, reloading applications, or other related tasks. You also need to mitigate

the attack vector that was used. Again, this can be a more difficult task than it appears, owing to the potentially incomplete situation surrounding the incident and what took place. You may discover that you cannot verify that backup media is clean, free of infection, or entirely corrupted.

### **Post-incident activity**

Post-incident activity, like preparation, is an easy phase to overlook, but you all must not do it. During the post-incident activity phase, also known as a post-mortem (Latin for “after death”), you try to figure out precisely what happened, why it happened, and what you can do to prevent it from happening. The goal of this stage is not to point fingers but to ultimately prevent or mitigate the impact of similar incidents in the future.

### **Defence in depth**

Defence in depth is a strategy used in military operations and information security. In both senses, the basic idea behind the defence in depth is to devise multi-layered defined lines or networks that will allow us to achieve a successful defence. Still, if one or more of our defensive measures fail, you would want defences at the external network, internal network, host, application, and data levels to work properly. Given well-implemented defences at each layer, users will make it extremely difficult to penetrate our network and attack our assets deeply. One important concept to note when planning a defensive strategy using defence in-depth is that it is not a magic bullet.

### **Activity 5.7: Analysing the vulnerabilities and creating a report on risk assessment**

**Resources:** A computer with Internet connectivity, networked computers, and a vulnerability-scanning tool.

#### **Procedures:**

- (a) Use any web browser to search for vulnerability-scanning tools
- (b) Download and install into your computer
- (c) Use the downloaded tools to scan the website and your network for vulnerabilities
- (d) Assess the risks and write a report about your observation
- (e) Choose the appropriate vulnerability-scanning tool to recommend in your computer Lab.

### **Authorization and access control techniques**

Authorization is a security method that evaluates access to the user or client privileges for system resources such as files, services, software programs, data, and applications. Likewise, it is the next step taken after completing identification and authentication.

### **Access control concepts**

Access control is a critical element of data security that governs who has access to and utilizes company services and data. Access control policies ensure users are said to have appropriate access to company data through authentication and authorization. But access controls are something you deal with daily. When you lock or unlock our house's

doors, you utilize a form of physical access control based on your keys. Moreover, computer system access control can be implemented in various ways depending on the situation. These include; locking your computer room and system, restricting system access to logins and passwords, protecting one's data with file protection or encryption, encrypting communication links, or validating a digital signature before accessing the data. Authentication mechanisms have two components of access control: authentication and authorization. An example of access control is when entering a server room or data centre using a physical key or fingerprint authentication, in which unauthorised users are denied access to restricted resources. Essentially, the user is prompted to provide a username and password when accessing computing resources. Another example is when the remote user is prompted to provide a username and password when accessing the network from outside the organization and is denied access while accessing confidential documents related to the company or a client.

### Types of access control

There are three kinds of access control such as system access, network access and data access

*Network Access:* Users on a network have access to all network resources. Consequently, network access should be restricted, protected, and monitored by users who can access the, for example, the Human Resources (HR) and finance divisions' LAN.

*System Access:* Users gain access to the network's systems. It can be one of the network's servers, printers, or shared devices. Access to these devices should be restricted, protected, and constantly monitored.

*Data Access:* Users are constantly accessing data from a network resource. Files, documents, and databases are made accessible and adapted by users. Any data that is accessed should be limited, protected, and monitored.

### Access control methodologies

Access control models can implement authorization and deny or grant access to parties based on which resources you determine they must have access to something. The term may sound highly technical and oriented toward high-security computing facilities.

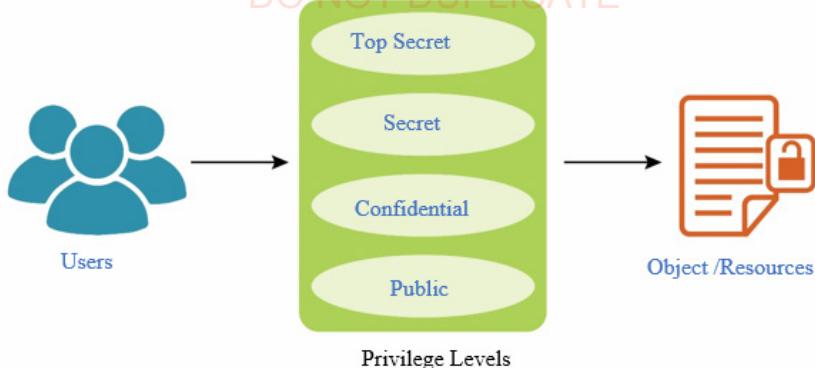
There are five access control methodologies which are: Mandatory Access Control (MAC), Discretionary Access Control (DAC), Role-Based Access Control (RBAC), Attribute-Based Access control (ABAC) and Multilevel Access Control.

#### *Mandatory access control (MAC)*

Mandatory Access Control (MAC) is a type of access control in which the resource owner does not have control over who has access to it. Still, a group or individual determines access with an authority to set access to resources.

Access to a given resource is frequently dictated by the sensitivity label (such as secret and top-secret), the level of sensitive information that the individual can access (perhaps only secret), and whether the individual needs to access the resource. Figure 5.14 shows Mandatory Access Control (MAC).

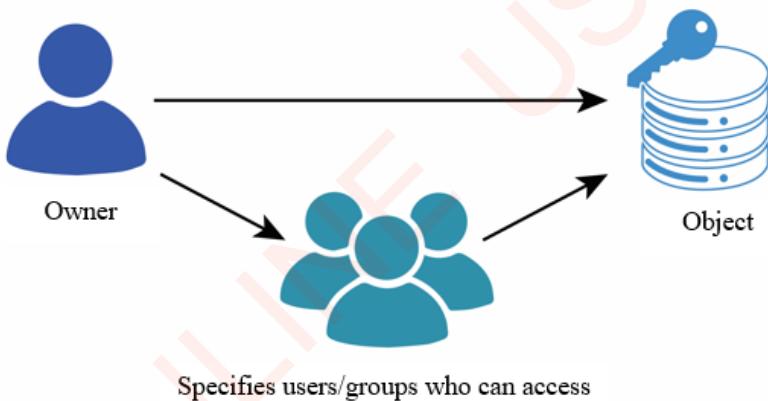
FOR ONLINE USE ONLY  
DO NOT DUPLICATE



**Figure 5.14:** Mandatory Access Control

#### *Discretionary Access Control (DAC)*

Discretionary Access Control (DAC) is a model of access control in which the resource's owner determines access. The owner can decide who has and does not have access to the resource and what level of access they are granted. We can see DAC implemented in Microsoft operating systems. If you decide to create a network share, you can choose who has access to it. Figure 5.15 shows DAC.



**Figure 5.15:** Discretionary Access Control (DAC)

#### *Role-Based Access Control (RBAC)*

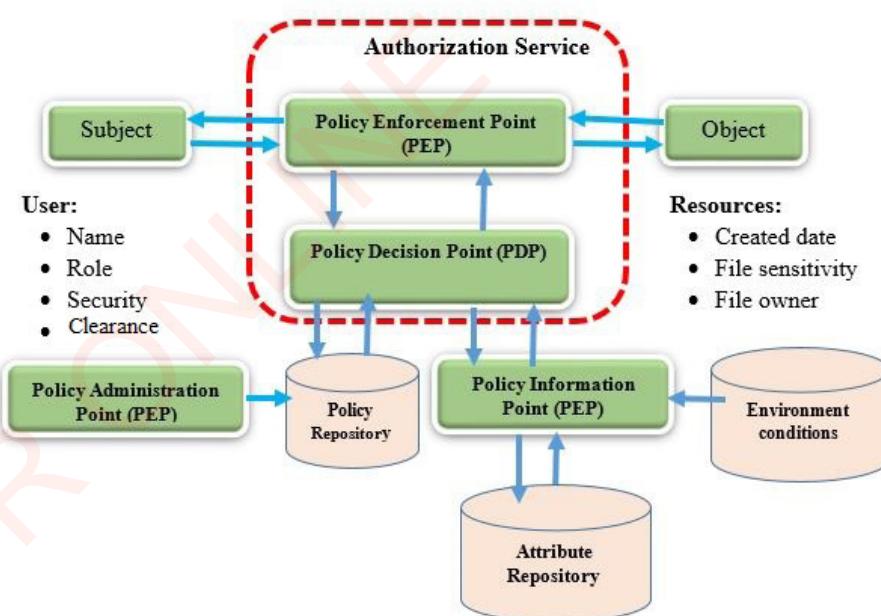
Role-Based Access Control (RBAC) is an access control model that, like MAC, is based on access controls described by authority rather than the resource owner. The distinction between RBAC and MAC is that access control in RBAC is based on the role of the individual being granted access. When you have an employee whose sole responsibility is to enter data into a specific application, RBAC once would only allow the employee access to that application, regardless of the sensitivity or lack thereof of any other resource he/she might potentially access. Suppose you have an employee with a more complex customer service role for an

online retail application. In that case, the employee's position could require the employee to have access to information about customers' payment status and information, shipping status, previous orders, and returns to assist customers.

#### *Attribute-Based Access Control (ABAC)*

Attribute-Based Access Control (ABAC) is logically based on attributes. These can be characteristics of a specific person, a resource, or an environment. The elements of a particular individual are referred to as the subject attributes. Humans can select from various features, such as the classic "you must be this tall to ride" access control, which exists to prevent the difficulties and challenges of riding on amusement park rides that may be hazardous to them. In theory, such an assignment might be difficult for a machine to perform since, it can not differentiate between human and robot

activities. Therefore, CAPTCHA is an acronym for Completely Automated Public Turing Test to Tell Computer and Human Apart, which is used to provide challenges that are difficult for the computer to perform but relatively easy for a human. CAPTCHAs prevent automated tools from performing tasks such as creating new accounts or posting spam comments on blogs. The attributes related to a specific resource, such as an Operating System or application, are resource attributes. When you encounter software that only runs on a particular operating system or websites that only work with specific browsers, we frequently see this happen, though usually for technical rather than security reasons. You could use this type of access control as a security measure by requiring the use of specific software or communication protocols, as shown in Figure 5.16.



**Figure 5.16.** Attribute-Based Access Control Mechanism

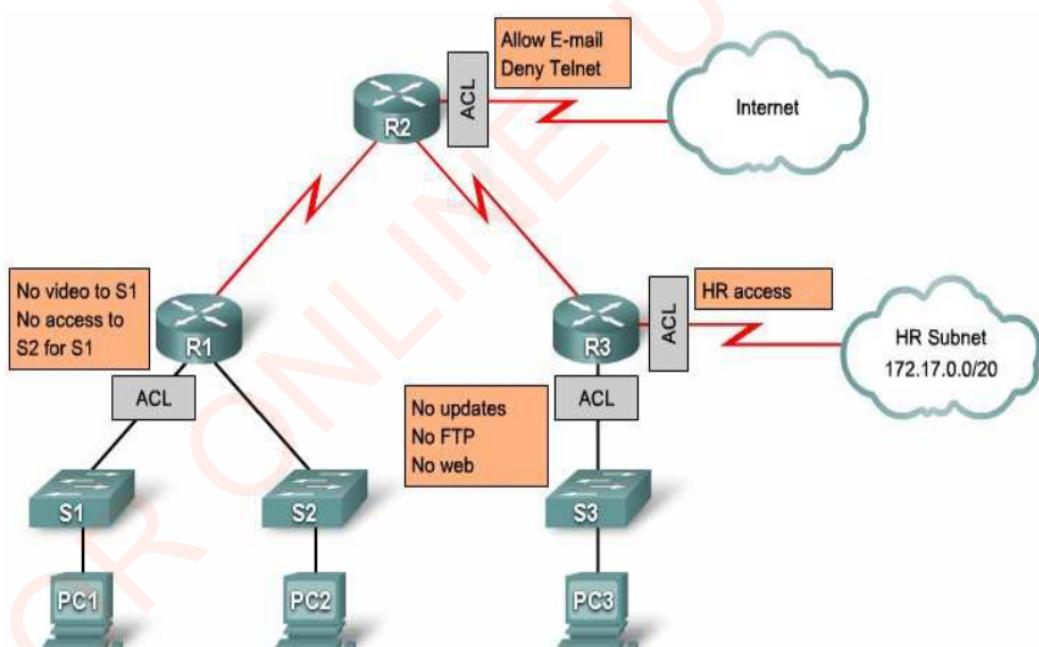
### Multilevel Access Control (MAC) NOT Data Access Control List (ACL)

Multilevel Access Control (MAC) models are used when simpler access control models are deemed insufficient to protect the information to which access is being controlled. Military and government organizations widely use access controls that frequently handle sensitive data.

Multilevel security models safeguard a wide range of data, from nuclear secrets to sensitive health information. For instance, suppose you have a secret resource and a user with such an intimate level of clearance, which typically allows them to access the resource under the accesses permitted by MAC.

An Access Control List (ACL) is a system of rules that specify which users' strategies are granted to access the target device or system resources. Access control lists are also installed in routers and switches, which serve as filters, determining which traffic is permitted to access the network.

Access control is typically used to control access in the file and operating systems and traffic flow in our systems' networks. ACLs are most frequently discussed in firewalls and routers. When ACLs are created, they are typically tailored to a specific resource, including identifiers of the parties permitted. Figure 5.17 shows the access control list configuration in various control equipment.



**Figure 5.17: Access Control list configurations**

**Functions of ACL**

1. Provide traffic flow control
2. Provide a basic level of security for network access
3. Decide which types of traffic are forwarded or blocked at the router interfaces
4. Allow an administrator to control what areas a client can access on a network
5. Limit network traffic and increase network performance
6. Screen specific hosts to either allow or deny access to part of a network
7. Deny user permission to access only certain types of files, such as FTP or HTTP

**Importance of access control**

- (a) It ensures that security technology and access control policies are in place to protect confidential information such as customer secrets and employee data.
- (b) Nevertheless, reduce the possibility of unauthorized access to physical and logical systems. Most businesses have infrastructure and procedures to limit access to networks, computer systems, applications, files, and sensitive data such as personally identifiable information and intellectual property.
- (c) It monitors the system and regulates by checking each person when entering and exiting the premises.

**Types of Access Control Lists (ACLs)**

The Access Control List involves two main types: File system ACLs and Networking ACLs.

**File System Access Control Lists**

File system ACLs filter access to files or directories and tell operating systems which users can access the system and what privileges are allowed. When looking at the ACLs in most file systems, you will notice three permissions in use, read, write, and execute, which will enable us to access the contents of a file or directory, write to a file or directory, and execute the file, presuming the file contains either a program or a script that can run on the system after execution. A file or directory in a file system may also be assigned multiple ACLs. For example, in UNIX-like Operating Systems, separate access lists for a given file can be seen in user, group, and other ACLs. You can grant an individual user or a group of different users read, write, and execute permissions. These three permission sets will be represented as rwxrwxrwx, with the first rwx permission set representing the user, the second the group, and the third the other. Furthermore, access permission for each file and directory in UNIX (and many other operating systems) is controlled by two identifiers: the User Identification Number (UID) and the Group Identification number (GID). Every user has a different username and belongs to at least one group. This data is retrieved from a password file. Only the administrator has the authority to implement or change a user's name and permissions. From the Linux or Ubuntu operating system, the *ls* command is one of the essential tools that lists files and directories with or without various

additional information, as illustrated in Figure 5.18.<sup>TE</sup>

```

File Edit View Search Terminal Help
tuts@ubuntu:~/Desktop$ ls -l
total 281280
-rw-rw-r-- 1 tuts tuts 765063 Feb 23 07:07 1wolf14.zip
-rw-rw-r-- 1 tuts tuts 280124238 Feb 24 04:12 BRL-CAD_7.26.0.2_x86_64.tar.gz
-rw-r--r-- 1 tuts tuts 204 Mar 17 08:54 'BRL CAD installation'
drwxrwxr-x 5 tuts tuts 4096 Feb 23 07:52 DOSBOX_Programs
-rw-r--r-- 1 tuts tuts 6769998 Mar 13 09:23 DSC_0627.JPG
drwxr-xr-x 2 tuts tuts 4096 Apr  8 02:27 FOSSLINUX
drwxr-xr-x 2 tuts tuts 4096 Apr  8 02:32 FOSSSTUTS
-rw-rw-r-- 1 tuts tuts 346719 Feb 23 07:44 prince-of-persia.zip
tuts@ubuntu:~/Desktop$ 

```

**Figure 5.18:** An example of the output that is produced by the 'ls –l' command

## Networking ACLs

Networking ACLs filter access to the network file Access Control Lists. Once users look at various activities on networks, both private and public, they will notice that ACLs are again regulating such activity. Access is typically controlled by the identifiers which humans use for authentication such as Internet protocol (IP) addresses, Media Access Control (MAC) addresses, other network transactions, and ports. ACLs can be seen in action in a network equipment/devices like Routers, switches, and firewalls, as well as applications such as, Facebook, Google, email, and other software. ACLs for network permissions are typically binary, composed of permit and deny. Finally, after configuring the ACL, you use your chosen identifier to specify which traffic you are referring to and simply state whether or not the traffic is

allowed. One of the most fundamental network-oriented ACLs you may encounter is MAC address filtering. In theory, MAC addresses are unique identifiers for every network interface in a given system. When a network interface is created, it is assigned a hard-coded MAC address.

### ***Classifications of network access control list***

Network access control lists can be classified as standard control lists and extended access control lists

**Standard ACLs:** Standard Control List filters traffic based on the source IP address and performs in sequence. Moreover, if placed closest to the source, it may deny or permit the host to access other resources in network. Example of the standard Access Control List configuration are illustrated in Figure 5.19(a) and (b).

```
access-list 10 permit 192.168.30.0 0.0.0.255
```

**Figure 5.19(a):** Standard ACLs configuration to filter IP packets based on the source address only

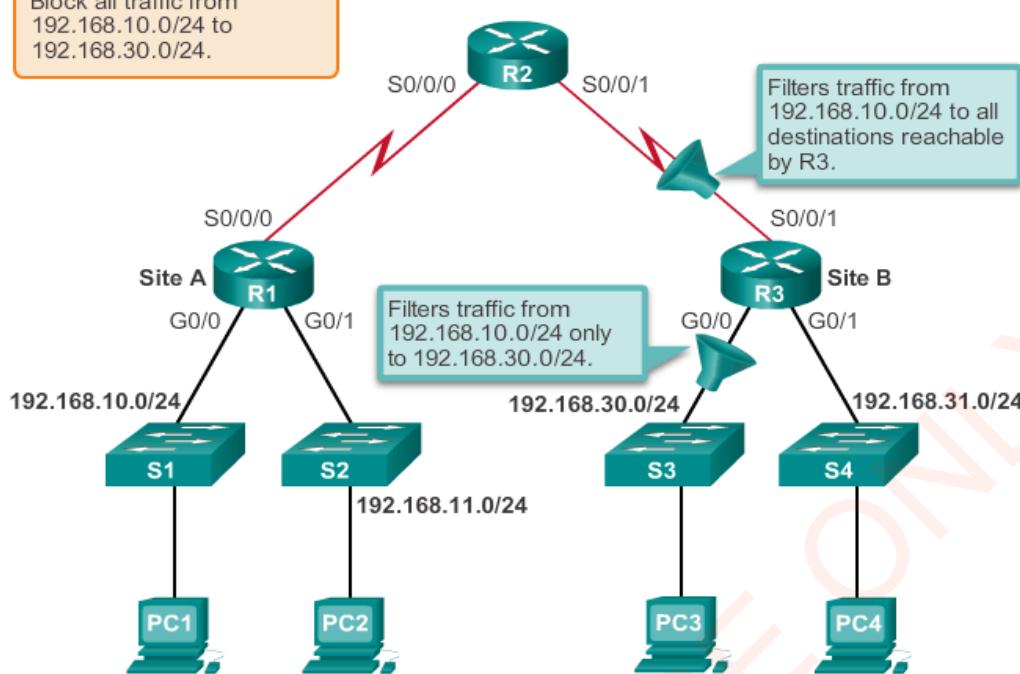
**Figure 5.19(b): Standard Access Control List (ACL)**

Figure 5.19(b) shows standard ACL configuration in which R2 is set to block all traffic from R1 in site A [192.168.10.0/24] which is intended to go to destination router 3 in site B [192.168.30.0/24]. Router 2 is also configured to filter traffic from R1 in site A to all destinations reachable through R3. Further, router R3 is configured to filter traffic from router R1 [192.168.10.0/24] intended for destination R3 [192.168.30.0/24].

#### *Extended ACLs*

Resembles the gatekeeper of the network that permits or denies traffic originating from any address on the given/ specified range. For example to permit traffic originating from any address on range 192.168.30.0/24, network to any destination host port 80(HTTP) can be configured through the following command:

```
access-list 103 permit tcp 192.168.30.0 0.0.0.255 any eq 80
```

Extended ACLs filter IP packets based on several attributes, including the following:

- (a) Source and destination IP address
- (b) Source and destination TCP and UDP ports
- (c) Protocol type/ protocol number (example: IP, ICP, UDP, TCP)

Figures 5.19 (c) shows the extended access control lists configuration.

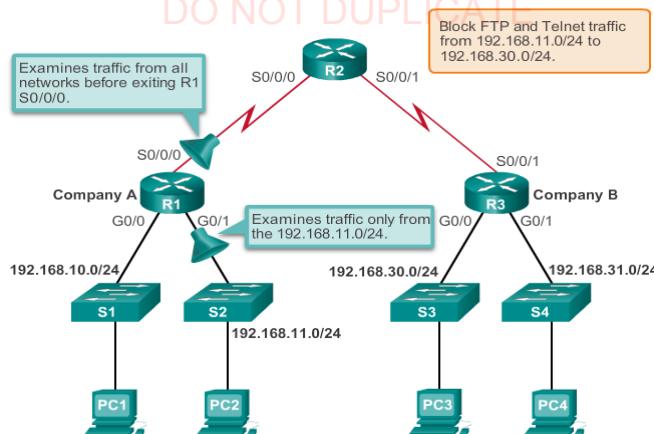
**Figure 5.19(c): Extended Access Control List configuration**

Figure 5.19(c) demonstrate the configuration of ACL in which router R2 insert filter traffic from all networks going out of R1. Further, R1 is also configured examine traffic coming from network 192.168.11.0/24. R2 is also configured to block FTP and telnet traffic which are coming from network 192.168.11.0/24 destined to network 192.168.30.0/24.

### ACLs operations

*Application of inbound and outbound access control list in packet filtering.*

An Access Control List operation is always an implicit deny. Ignoring the fact that it is not physically present, this statement is automatically inserted at the end of each ACL. The implicit denial shuts down all traffic. An ACL that lacks at least one permit statement will block all traffic due to this implicit denial.



**Activity 5.8:** Download Cisco packet tracer tools (Network simulator) from the internet and draw the sketch map of any topology in your school lab. After that, configure and test ACLs as per topology.

### Network access control tools

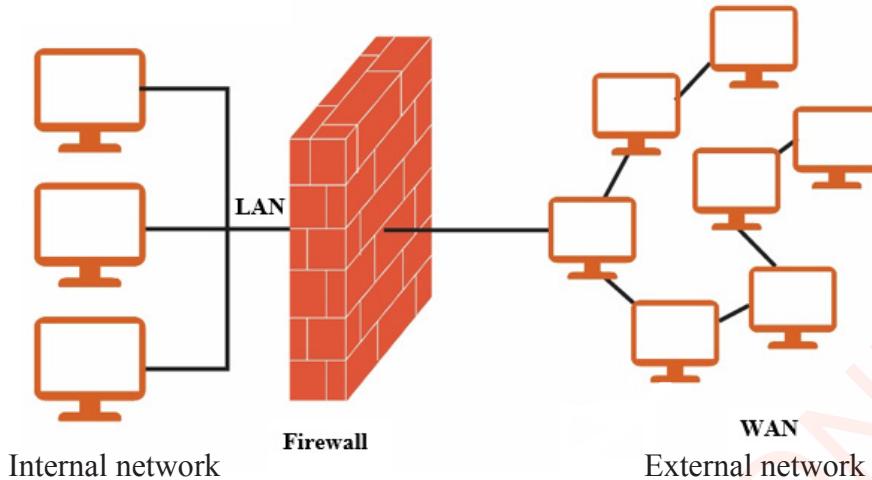
There are several network access control tools as described in the next paragraph:

#### Firewall

Firewalls are devices or programs that regulate network traffic flow between networks or hosts with varying security postures. While firewalls are frequently discussed in terms of Internet connectivity, they may also be helpful in other network environments. For example, many enterprise networks use firewalls to restrict connectivity to and from internal networks that serve more sensitive functions, such as accounting or personnel. An organization can prevent unauthorized access to its systems and resources by using firewalls to control connectivity to these areas. The addition of an appropriate firewall adds an added level of security.

In particular, firewalls are needed to prevent intruders from entering and interfacing with the operations of an organization's network system, deleting or modifying information stored or

in motion within the organization's network system and acquiring proprietary information. Figure 5.20 illustrates Firewalls.



**Figure 5.20: Firewalls**

### Types of firewalls

There are two types of firewalls: packet filtering and proxy firewall.

Apart from that, there are variations of two types of firewalls, commonly called gateway or bastion.

#### Packet filtering firewall

A **packet filter firewall** is a multilevel firewall that compares and filters all inbound and sometimes outgoing network traffic passing through it. It contrasts those packets to a predefined set of rules. The firewall accepts the packet that matches a rule. A package is rejected or logged for further investigation if it does not match a rule.

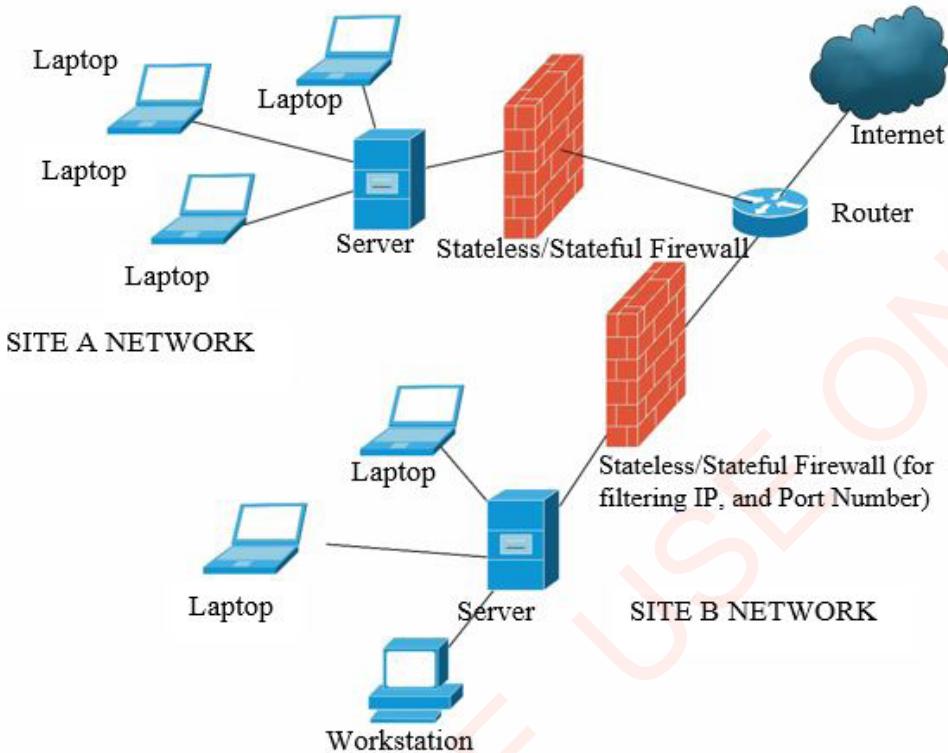
Additional data packet screening may be required for further investigation, in which case the firewall directs the datagram to the screening device. Following further screening, the datagram may be allowed through or denied. Many filter firewalls use protocol-specific filtering criteria at

the data link, network, and transport layers. The firewall compares data in each datagram at each layer, such as the source and destination addresses (The type of service requested and the type of data delivered). A data packet is denied, accepted, or delayed based on one or more conditions: source address, a destination address, TCP or UTP source and destination port, ICMP message type, Payload data type, Connection initialization, and datagrams using TCP ACK bit.

Moreover, a packet filter firewall is itself divided into two configurations. One is a straight packet filter firewall, which allows full-duplex communication. This two-way communication is made possible by following specific rules for communication application proxy. Travel traffic in each direction for each datagram is examined for the particular criteria given above, and if conformity to direction-specific rules is established, the firewall lets the datagram through.

The second configuration is the stateful inspection packet filter firewall, a full-duplex filter firewall; however, it filters using a more complex set of criteria that involve more restrictions than those used by a straight packet filter firewall.

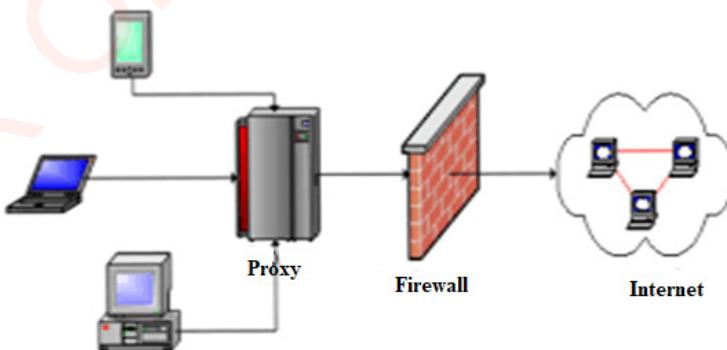
These harsh restrictions form a group of one-way rules for the stateful inspection filter firewall. Figure 5.21 shows the packet filter firewall.



**Figure 5.21:** A Packet Filter Firewall

### Proxy firewalls

A proxy firewall is a network security system that safeguards network resources by filtering messages at the application layer. A proxy firewall is also referred to as an application firewall or a gateway firewall. Figure 5.22 illustrates the proxy firewall.



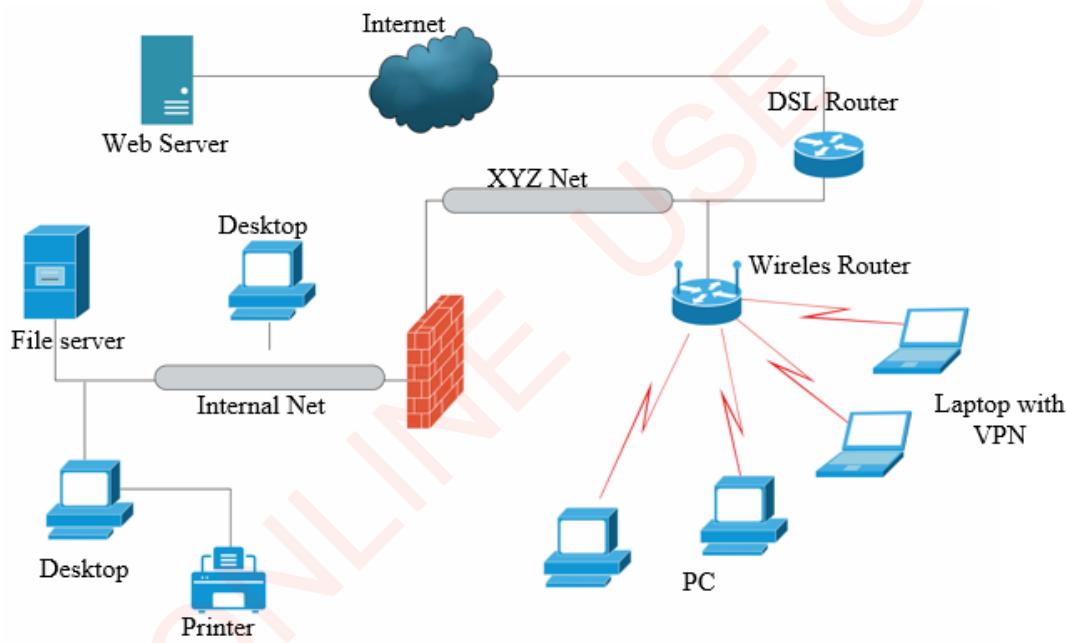
**Figure 5.22:** Proxy firewall

Proxy firewalls are divided into two types: the Application-level proxy firewall and the Circuit-level proxy firewall.

#### *Application-level proxy firewalls*

A firewall generates a proxy for each application developed by a communicating element. The proxy firewall inspects and forwards each application-generated traffic. The presence of an application proxy makes logs and controls incoming and outgoing traffic, offering a higher degree of security such as, user-level authentication, end-to-end encryption, intelligent logging, information hiding, and access restrictions based on service types. A proxy filter firewall is shown in Figure 5.23.

Internal networks like LANs usually have multiple application proxy firewalls, including telnet, WWW, FTP, and SMTP (email). Although application proxy firewalls are great as high-level filtering devices; they are more expensive to install because they may require installing a proxy firewall for each application an organization has, and that can be expensive to acquire, install, and maintain.



**Figure 5.23:** Application Proxy Firewall

#### **Circuit-level proxy firewall (with low-level general protocol proposals)**

This type of proxy firewall treats all network packets as if they are a slew of black boxes to be routed through a filter or a bastion or not. It only filters based on the information in the packet header. As a result, it is faster than its cousin, the application-level proxy.

A gateway is commonly known as a bastion gateway for a combination of the filter and proxy firewalls, which gives it a medieval castle flavour. Packets from the local network and outside the Network can only reach their destinations through a firewall by passing through the filter router and proxy by the station. Each application gateway combines a general-purpose router that acts as a traffic filter with an application-specific server that all application data must pass through.

### Function of firewall

The following are the functions of firewall:

- As a network security guard. It filters all traffic that enter and leave the network according to the predefined roles, mechanisms and policies.
- Prevention of information leaking without knowledge. Through a set of protocols, such as, FTP it can detect and prevent users from sending valuable data to other parties.
- Keeping records of user activities.

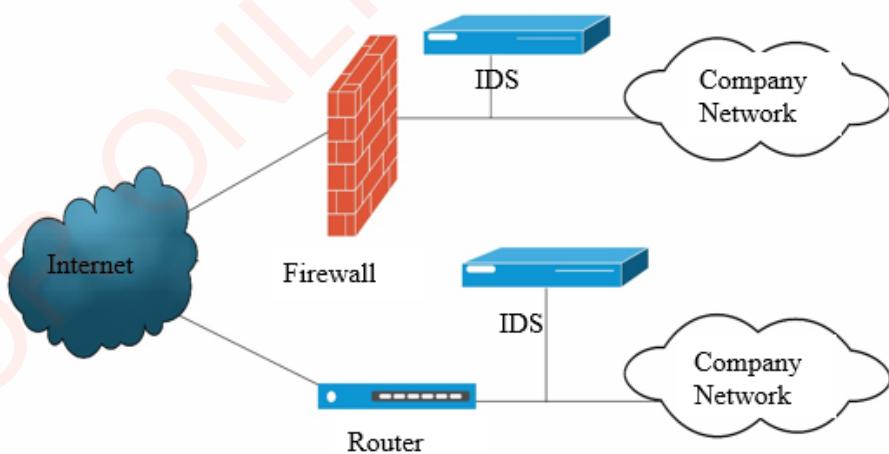
Every user who access the network, will go through the firewall which record it as a log file to be used later for statistics on network usage.

- Prevention of modification of valuable information.

### Intrusion detection and prevention system

#### *Intrusion detection*

This refers to software or device which monitors the network traffic for malicious activity or policy violations. Detecting intrusions is based on models and patterns of what something good looks like (typically built by normal baseline behaviour) and looking for deviations/anomalies. These systems learn primarily by observing our networks and systems in use and then simply having a much higher knowledge about the traffic. From this point of view, IDS can reclaim attention when combined with a policy enforcement system such as MAC to take action against wrong sources of malicious traffic. Figure 5.24 depicts the intrusion detection system and possible locations.



**Figure 5.24: Intrusion Detection System (IDS)**

### *Types of intrusion detection systems*

Typically, there are two types of IDS which are Host-based IDS and Network-based IDS/IPS

#### *Host-based IDS*

Host-based Intrusion Detection System is typically a software-based deployment IDS in which an agent is installed on the local host to monitor and report on application activity, as depicted in Figure 5.25, HIDS monitors the system and access privileges and sends alerts for any unusual activity. It continuously monitors event logs, system logs, application logs, user policy enforcement, rootkit detection, file integrity, and other computer system intrusions.

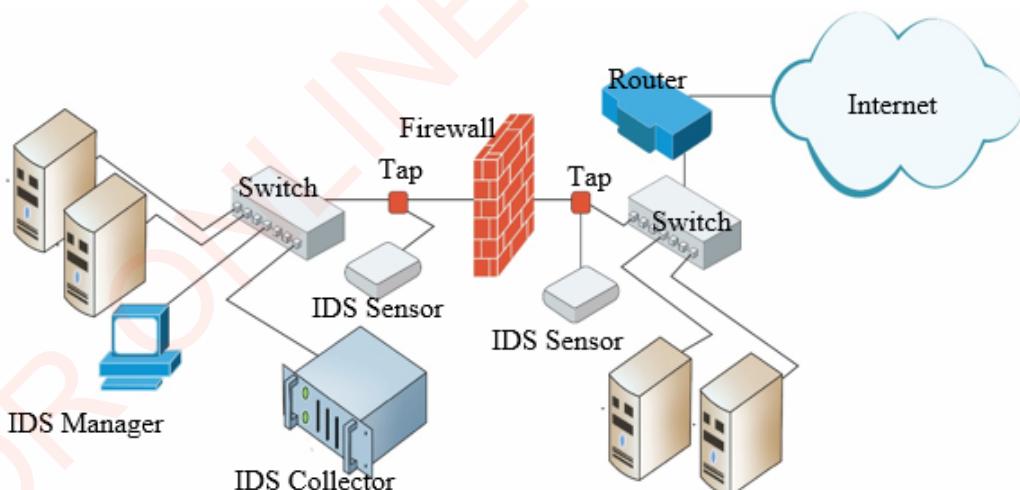
It continuously monitors these logs and establishes a baseline. If any new log entries appear, HIDS compares the data to the baseline, and if any entries are found outside of the baseline, HIDS generates an alert. If any unauthorized

activity is detected, HIDS can alert the user, block the action, or create any other decision based on the system's policy.

It should be capable of detecting both known and unknown attacks. The majority of HIDS products can also prevent attacks.

Nevertheless, it is initially deployed in the monitor mode, and once the system activity is understood, a baseline is created, and HIDS is deployed in the prevention mode. The functionality of HIDS depends on the records and the fact that attackers leave indications of their activities.

In general, hackers seek to gain access to the system and install malicious tools to facilitate future access. If these tools alter the operating system configurations or registry entries, it is logged in the systems/event log, triggering an alert by the HIDS system. Figure 5.25 shows a host-based intrusion detection system.

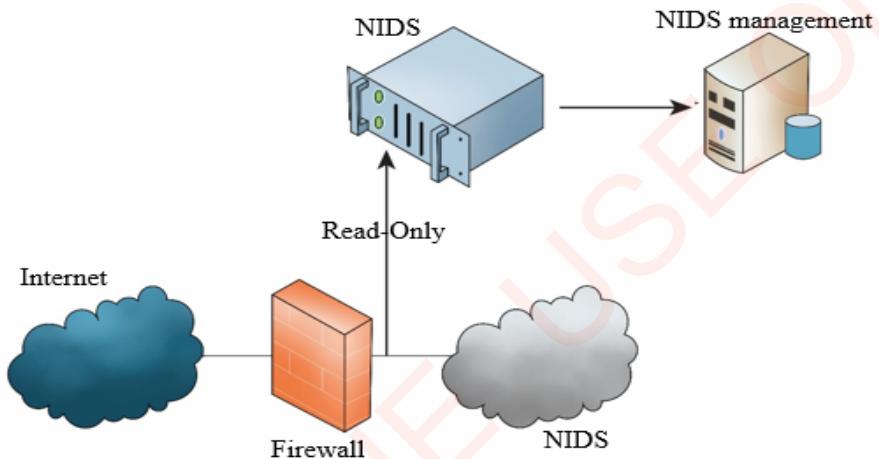


**Figure 5.25: Host-Based Intrusion Detection System**

## Network-based IDS and Intrusion Prevention System (IPS)

Network intrusion detection systems (IDS) and intrusion prevention systems (IPS) protect the network and its resources from a network malicious activity standpoint. For example, network intrusion detection systems (IDS) can detect reconnaissance and denial-of-service attacks at the network level. As soon as these threats are detected, NIDS generates alerts. NIDS is a hardware/software solution that operates as an independent device (sensor) near the firewall and has a network operating system (TCP/IP stack). Sensors have interfaces for monitoring the network (monitoring interfaces), controlling and receiving alerts, and sending these alerts to the central management console. Figure 5.26 depicts network-based intrusion detection and prevention system.

IDS/IPS is another network-based intrusion detection which employs signature-based and anomaly-based detection techniques.



**Figure 5.26: Network-Based Intrusion Detection and Prevention System**

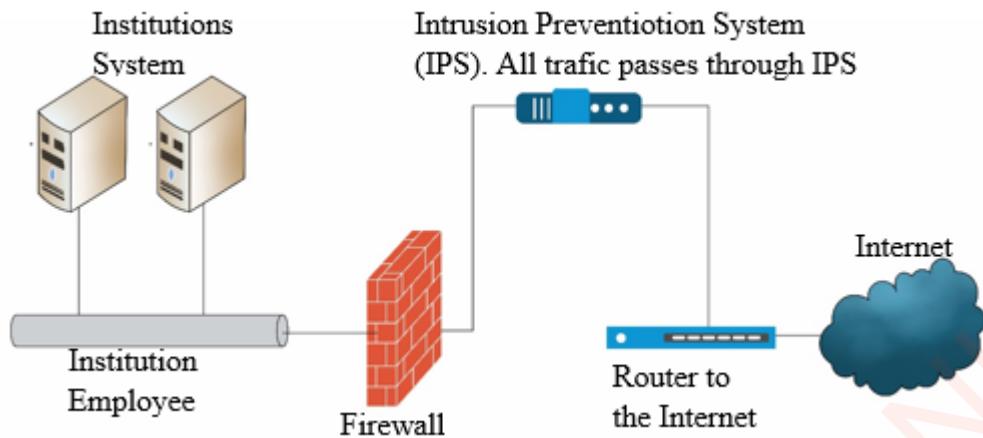
Intrusion detection systems employ two detection methods: Signature-based detection and behaviour-based detection.

*Signature-based detection* compares data activity to a signature or pattern in a signature database. For example, signature-based detection has a constraint that ignores new malicious activity that is not in the database.

*Behaviour-based detection* is the one which detects any anomaly and provides alerts, implying that it can see new types of attacks. It is known as an expert system because it learns what normal behaviour in your system appears equivalent to a network.

An Intrusion Prevention System (IPS) is crucial for network security. It is based on IDS (Intrusion Detection System) as the extra step of taking action, often in real-time, to prevent an intrusion once detected by the IDS. So, Protection = Prevention plus

Detection. Detecting attacks is a fundamentally different problem from preventing intrusions. Figure 5.27 shows Intrusion Prevention System (IPS).



**Figure 5.27:** Intrusion Prevention System (IPS)

### Intrusion Prevention System (IPS)

The Intrusion Prevention System (IPS) is used to prevent intrusion. Advancement to IDS. IDS only detect whereas IPS protects the network from intrusion by dropping packets, denying message entry, or blocking the link. IPS and IDS work together to monitor network traffic for threat actors, and IPS is simply a part of IDS. The main difference is that the IPS is placed in-line to prevent intrusions. The IPS can make decisions, including dropping packets or resetting interconnection and sending alarms to the management console. An Intrusion Prevention System (IPS) can detect and correct fragmented packets, Cyclic Redundancy Check (CRC) errors, and TCP sequencing issues.

IDS operation on system and network (how it operate on the system and Network)

### Virtual Private Network (VPN)

The ability to encrypt and decrypt specific network traffic flows between the protected network and external networks is a common requirement for these firewalls. Virtual Private Networks (VPNs) are almost always used, which use additional protocols to encrypt traffic and provide user authentication and integrity checking. VPNs are most commonly used to secure network communications across untrusted networks. VPN technology, for example, is widely used to extend the protected network of a multi-site network. Apart from that, it is a private network that connects to remote sites via the public network.

VPN sets up a “virtual” tunnel connection via the Internet from the company’s trusted network to a remote office or mobile workforce. VPN allows you to send data over a public network while simulating a private link between two parties of the Network.

## Types of VPN

Virtual Private Network (VPN) is divided into two types: Remote Access VPN and Site to Site VPN (Intranet and Extranet); however, there is another type called Host-to-Host VPN.

### Remote Access VPN

Remote Access VPN allows users to connect to a private network and remotely access most of their resources and services. The user's connection to the private network is established via the Internet, and the link is safe and confidential. Remote Access VPN can be used for various purposes, at home and office users. Second, the company employee uses a VPN to connect to their company's private network and remotely access files and resources on the private network. VPN users who are remote or at home use VPN services to bypass regional Internet restrictions and access restricted websites. Internet security-conscious users use VPN services to improve their online security and privacy.

### The Site to Site VPN (Intranet and Extranet)

A Site-to-Site VPN: Site-to-Site VPN is also called a Router-to-Router VPN and is commonly used in large companies. Companies or organizations with branch offices in different locations use a Site-to-Site VPN to connect the Network of one office location to the network at another office location.

*Intranet-based VPN:* When several offices of the same company are connected using a Site-to-Site VPN, it is called an Intranet-based VPN. On the other hand.

*Extranet based VPN:* is when companies use a Site-to-Site VPN type to connect to the office of another company.

### Host-to-Host VPN

With one minor exception, some people refer to host-to-host VPNs as remote access. A VPN tunnel that connects two hosts is a host-to-host VPN. The tunnel is established directly between two hosts for secure data transfer. The user is authenticated before the data transmission, and the encrypted credentials are exchanged between the two parties before data transmission. The VPN tunnel protects data by ensuring its authenticity, integrity, and confidentiality. This type of connection is permitted when an employee or a partner wishes to securely connect to a specific network resource (server/database). An individual person may not be allowed to access any other network resources.

## Classifications of VPN

Two common choices for secure VPNs are classified as IPsec and Secure Sockets Layer (SSL)/Transport Layer Security (TLS).

### Internet Protocol Security (IPsec)

Internet Protocol Security, known as IPsec, is used to secure Internet communication across an IP network.

### **Secure Sockets Layer (SSL) and Transport Layer Security (TLS):**

Secure Sockets Layer (SSL) and Transport Layer Security (TLS) generate a VPN connection where the web browser acts as the client. User access is prohibited to specific applications instead of the entire network. Online shopping websites commonly use SSL and TLS protocols. It is easy to switch to SSL by web browsers and with almost no action required from the user as web browsers are integrated with SSL and TLS. SSL connections have “https” in the initial of the URL instead of “http”.

### **Secure Shell (SSH)**

Secure Shell generates the VPN tunnel through which the data transfer occurs and ensures that the tunnel is encrypted. An SSH client causes SSH connections, and data is transferred from a local port to the remote server through the encrypted tunnel. Figure 5.28 shows Secure Shell (SSH).



**Figure 5.28: Secure Shell (SSH)**

### **VPN protocols (IPsec, L2TP, PPTP, GRE)**

VPN protocols include IPsec, PPTP, L2TP and GRE. IPsec protects Internet Protocol communication by validating the session and encrypting each data packet sent during the connection. Furthermore, IPsec’s goal is to provide security services for the IP layer in both IPv4 and IPv6 protocols. IPsec is a cryptographically designed security protocol for the IPv4 and IPv6 protocols. IPsec security services protect network traffic by ensuring data confidentiality, integrity, authenticity, and anti-replay.

IPsec is a complex framework of numerous protocols that provide various security features. To accomplish this, IPsec employs two major security protocols: the Authentication Header (AH) and the Encapsulating Security Protocol (ESP).

The IPsec framework has two modes, namely:

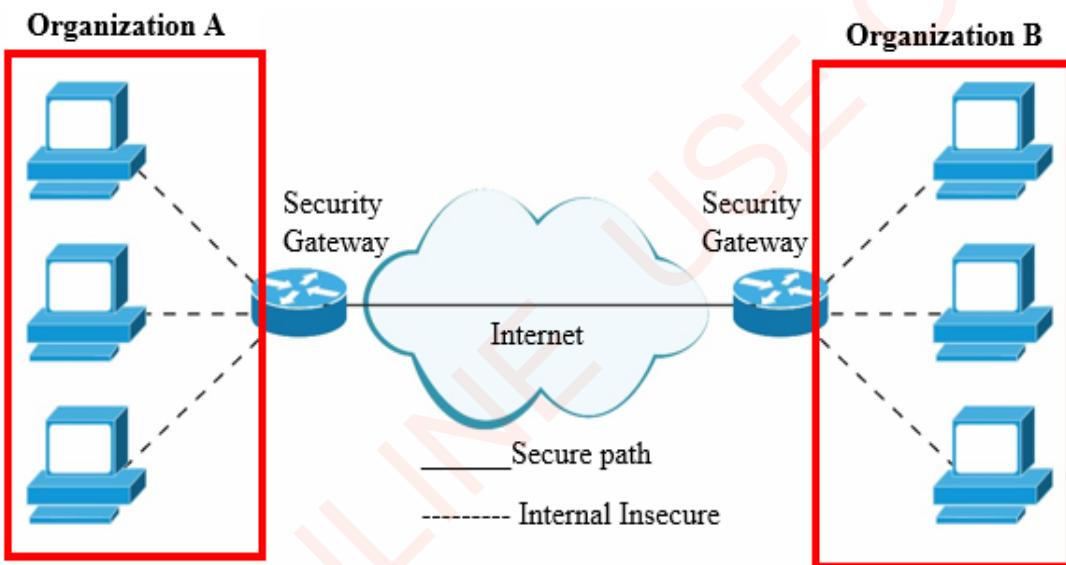
- (a) Transport mode
- (b) Tunnelling mode

#### **Transport mode**

The transport mode encrypts the message within the data packet, while the tunnelling mode encrypts the entire data packet. IPsec can also be used in conjunction with other security protocols to boost the security system.

#### **Tunnelling mode (L2TP, or Layer 2 Tunnelling Protocol)**

Layer 2 Tunnelling Protocol is a tunnelling protocol frequently used in conjunction with another VPN security protocol, such as IPsec, to create a highly secure VPN connection. L2TP creates a tunnel between two L2TP connection points, and the IPsec protocol encrypts the data and helps ensure encrypted communications across the tunnel. Figure 5.29 shows the Layer 2 Tunnelling protocol.

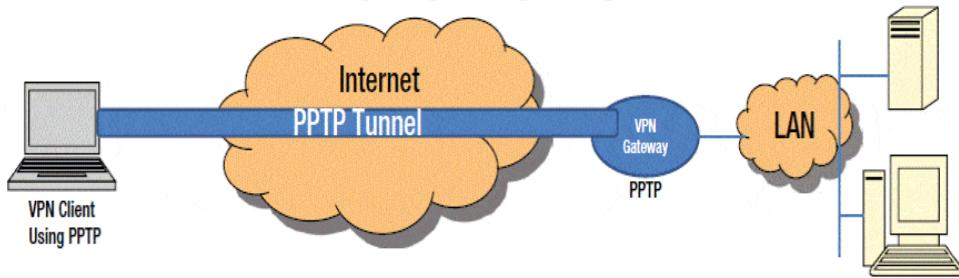


**Figure 5.29: Layer 2 Tunnelling Protocol**

#### **Point-to-Point Tunnelling Protocol (PPTP)**

The Point-to-Point Tunnelling Protocol creates a tunnel and confines the data packet. The Point-to-Point Tunnelling Protocol (PPTP) is used to encrypt data between connections. It is a popular Hosting protocol that has been in use since the early days of Windows. Figure 5.30 shows the Point-to-Point Tunnelling protocol.

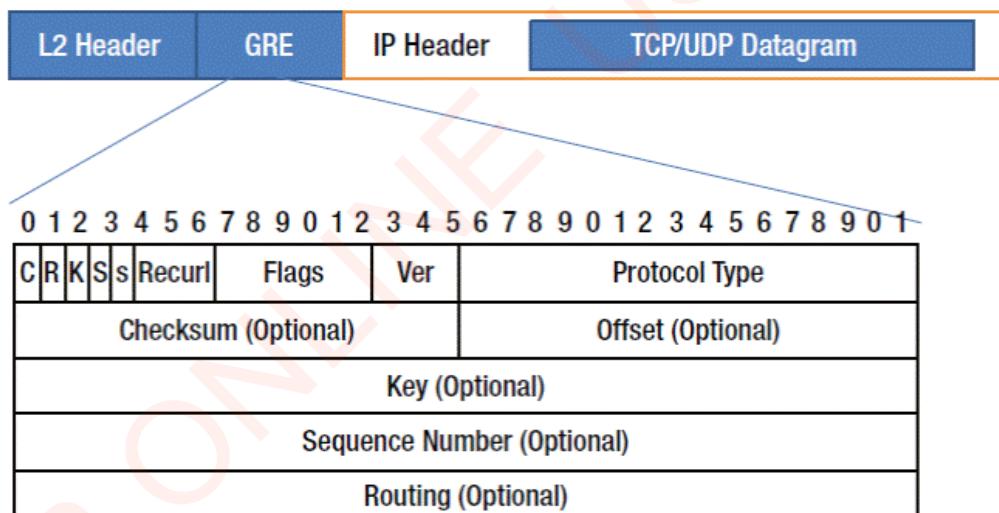
FOR ONLINE USE ONLY  
DO NOT DUPLICATE



**Figure 5.30:** Point-to-Point Tunnelling Protocol

### Generic Routing Encapsulation (GRE)

The tunnelling protocol Generic Routing Encapsulation (GRE) encapsulates one IP datagram within another IP datagram and transports the encapsulated IP datagram. GRE, in other words, encapsulates one network layer protocol inside another network layer protocol. RFC 2890.6 describes the general specification of the GRE. Figure 5.31 depicts a typical GRE diagram. A network layer packet, known as the “payload” packet, is encapsulated in a GRE packet, which may include all the network payload packet’s routing information. The resulting GRE packet is then encapsulated in another network layer protocol called the “delivery protocol.”



**Figure 5.31:** Format of a GRE encapsulated packet



### Activity 5.9: Confirmation of access control list

You want to apply an access list of 198 to an interface to filter traffic into the interface. Write a command that you will achieve in this confirmation of access control list.

**Exercise 5.2.**

1. Discuss with vivid examples two types of access control.
2. Distinguish between Extended ACL and Standard ACL.
3. Write brief notes on firewalls with vivid examples.
4. Differentiate between layer 2 tunnelling protocol from point to point tunnelling protocol.

**Data loss prevention**

Data loss prevention (DLP) involves a collection of tools and processes used to keep sensitive data from being lost, misused, or accessed by unauthorized users. DLP software categorizes regulated, confidential, and business-critical data and detects violations of policies defined by organizations or acts as a check pack, which are typically driven by regulatory compliance such as HIPAA, PCI-DSS, or GDPR.

**Importance of data loss prevention**

- (a) DLP helps to prevent the involuntary disclosure of sensitive information across all devices. DLP can monitor data wherever it is, whether in transit on the network, at rest in backups, or in use, and reduce the danger of data loss.
- (b) DLP can identify, classify, and tag sensitive data and monitor the activities and events that occur in the context of that data.
- (c) It produces a report of the information required for compliance audits. IP Protection: Does your organization have valuable intellectual property and trade or state secrets that, if lost or stolen, affect financial health and

brand image? DLP solutions that use context-based classification, such as Digital Guardian, can classify intellectual property in structured and unstructured forms. You can prevent unwanted data loss or theft by implementing policies and controls.

- (d) Data Visibility: Does your organization want to gain more insight into data movement? A cohesive business DLP solution can assist you in seeing and tracking your data across endpoints, networks, and the cloud.
- (e) It gives you insight into how your organisation's users interact with data. While these are the three most popular use cases, DLP can also help with insider threats, Office 2019 data security, user and entity behaviour analysis, and advanced threats.
- (f) DLP software scans incoming emails for malware or suspect connections. These tools enable you to flag inconsistent content so that employees can manually process, assess it or stop it in its tracks once it is detected.

**Drawbacks of data loss prevention**

Encrypting network traffic has two significant drawbacks. For beginners, network encryption has been challenging to implement in past centuries. Using IPsec encryption, for example, usually requires the setup of an enterprise certificate authority. An administrator must still understand the essential management process and how to set group policies requiring network computers to use IPsec encryption. Furthermore, IPsec encryption will fail unless network clients use IPsec-compatible operating systems.

Another significant drawback of network traffic encryption is that it can degrade performance. Every time a client needs to communicate over the network, the client must establish a session and encrypt the data to be sent. The recipient must decrypt the data. This process increases the amount of network traffic and requires network client machines to spend more time and CPU resources encrypting and decrypting data. There should be network cards that can take the encryption and decryption process away from the CPU. Hence, it prevents network clients from undergoing poor performance.

### Exercise 5.3

1. What is Data Loss Prevention (DLP)?
2. Type an E-mail message to your teacher asking for test two (2) results. Apply the skill you have mastered in the session to encrypt email and show how a message you sent was encrypted.
3. When someone tries to send an encrypted message to your E-mail, what does an encrypted message look like for the external user?

### Data backups and disaster recovery

Data backup is the periodic storing of copies of copied files regularly and keeping them in a safe place or a safe storage device.

#### Importance of data backup and disaster recovery

When files have been restored, they can be reorganized to occupy contiguous

space, resulting in faster access. On site and off site storage is critical to a fire-proof safe where in case of any damage a copy can be taken from on-site or off-site to restore. A benefit is that files that have become fragmented can be reorganised to occupy contiguous space when restored, resulting in quicker access.

#### Types of data backups technologies

*Digital linear tape (DLT) or Linear tape-open (LTO) systems:* These are rock solid which helps to address storage capacity and access time. They also, increase the total capacity and speed of the tape and disk when information is backed up into the tape.

Robotic auto-changers are available for DLT, and LTO drives with plenty of headroom and can hold five tapes up to large libraries that hold tens or hundreds of videos.

*DLT S4 (800GB per tape) and LTO – 6 (2,500GB per tape):* These are very fast and reliable with high per tape capacities. Examples: Tape Vaulting, Remote Tape Backup, and Offsite Backup.

**Cloud Storage:** Cloud storage involve storing data to online servers located and hosted somewhere else. This backup solution is unique because you can back up all your local files and securely store them in your chosen cloud provider. Caution on security should be considered because the data are stored in a third party.

Even if your data centre experiences a catastrophic event, your hosted data and territories will be preserved when using this type of backup.

## Data backup requirements

- (a) Keeping backups safe; backup copies need to be held in a fireproof safe or preferably offsite
- (b) Data backups restorations strategies
- (c) Keep the daily backups for a week
- (d) Keep Fridays back up for a month
- (e) Keep one backup each month for a year
- (f) To prevent mix-ups, give each tape or disk a serial number and keep a logbook

## Data recovery mechanism

Testing recovery procedures and the effectiveness of backup systems need to be done regularly. It helps to ensure that a business can recover from a disaster.

An additional contingency plan must be developed to consider alternative compatible equipment and security facilities, including temporary office space provision of alternative communications links.

## Fault tolerance mechanisms

Fault tolerance is the ability of a system like a computer, network, cloud or cluster to continue operating without interruption when one or more of its components stop working. It includes a server cluster, remote journaling, and server mirroring to secure situations.

### **Server clustering**

A server cluster is a group of servers that work collaboratively on a single system to provide users with increased availability. These clusters mitigate downtime and outages by allowing some other server to take over in case

of an outage. A collection of servers is linked to a single system that acts as a stand-by when it fails.

Clustered servers are typically used for applications that require frequent data updates, with file, print, database, and messaging servers being the most common clusters. Overall, clustering servers provide clients with higher availability, reliability, and scalability than any single server could. In a clustered hosting environment, each server is responsible for the owners and control of its own devices and for getting a copy of the operating system (along with any applications or services) used to run the other servers in the cluster. Electronic Vaulting means that data are backed-up. The output is electronically transmitted to a secured offsite storage location. Examples of server clusters are Tape Vaulting, Remote Tape Backup, and Offsite Backup.

### **Remote journaling**

Remote journaling and electronic vaulting are mainly used in systems where data is a priority, ensuring that not even a tiny section of the information is changed or lost. It is the process of recording transactions at remote locations as a backup strategy. It allows the user to capture the occurrence of changes and then copy them to a different location. It also keeps changes in data since the previous save. Recovering data requires replaying the changes and cleaning up any changes.

### **Server mirroring**

Server mirroring is a network control process that continuously creates an exact copy of a server. Server mirroring

is a method for ensuring business continuity, disaster recovery, and backup. Furthermore, it is primarily used to build a fault-tolerant and redundant server computing infrastructure. This process is carried out by using specialized backup software installed on the backup server and the server to be mirrored. The software routinely synchronizes and backs up data from the primary server to the backup server via a secure Internet connection or VPN.



### Activity 5.10: Creating backups

**Resources:** Sample files, a computer, external storage devices such as HDD, memory sticks, internet connectivity and an active yahoo or Gmail email address, a computer and a smartphone.

**Procedures:**

1. Open your computer and access the internet
2. Create an e-mail account if you don't have one
3. Install and configure any cloud storage tools such as Dropbox
4. Create a folder on your computer and link it to your cloud storage
5. Save your file in the folder on your computer and then
6. Use the smartphone to see if you can access the file from such a computer.
7. Make several copies of your file into the storage devices you have
8. Try to think about what will happen if your flash is lost or corrupted
9. Share with your colleague and then to the class and suggest the best option for your class.

## Vulnerability analysis and penetration testing

A vulnerability analysis is a review that focuses on threat protection issues that have a moderate effect on the system's security.

### Penetration testing through hacking

Penetration testing is a method of testing the security of a network and the vulnerability of a company's systems and infrastructure by employing various tools and techniques used by attackers. The approach is similar to an attacker: enumerate the network, assess vulnerabilities, research vulnerabilities for known exploits, and then use methods available to penetrate the network. The attackers list a grid to decide which machines are connected and operationally useful for both an intruder and a system administrator. The data obtained from a network scan aids in determining the actual current layout.

A few tools and techniques are available for both the Windows and Linux platforms to perform these tests. After identifying the devices and their open ports, a vulnerability scanner can be used. The scanner will use its vulnerability database to see if the system has any vulnerabilities. These vulnerabilities are then researched further online, and utilities that can be used to breach the network are downloaded and executed. A good penetration test should produce a report explaining the weaknesses discovered, ranking them from most critical to least critical, and offering suggestions for improving network security.

## Importance of penetration testing

Penetration tests are essential for an organization's security because they teach employees how to deal with any malicious entity break-in. Penetration tests are used to determine whether or not a company's security policies are effective. Penetration testing and vulnerability analysis are critical network security components, serving different purposes. Penetration testing simulates a real-world attack on a network's defences. On the other hand, a vulnerability is a non-intrusive scan that searches the web for potential vulnerabilities.

One of the most important things to consider when starting a new business is regulatory compliance. The regulatory aspect is one of the most important considerations for any business's success. Every industry has its own set of regulations and rules.

It helps in identifying vulnerabilities in target applications. Businesses and organizations commonly use it to ensure compliance with government regulations.

## Hacking

Hacking compromises digital devices and networks by gaining unauthorised access to any account or computer system. Exploiting vulnerabilities in a system, compromising security, and gaining unauthorized control and authority over the system's resources are all examples of hacking.

### Types of hackers

There are many types of hackers, such as cyber terrorists, state or nation-

sponsored hackers, black hat hackers, white hat hackers, grey hat hackers, Script Kiddies, red hat hackers, hacktivists, and malicious insiders or whistle-blowers.

### Cyber terrorism

Cyberterrorism (also known as digital terrorism) is a disruptive attack on computer systems by recognized terrorist organizations with the intent of causing alarm, panic, or physical disruption of the information system. Examples are the disorder of major websites, the attack on water treatment information systems, the interruption of an oil pipeline, (cyber espionage), and the disruption of systems and networks.

### Black hat hackers

These individuals, also known as criminal hackers or crackers, maliciously gain access to another person's system for personal gain. They commonly hack electronic devices and modify, steal, or delete critical files for personal growth.

### White hat hackers

White hat hackers, also known as ethical hackers, discover ways to exploit a device's system to learn how people can defend themselves against potential attacks. These ethical hackers also ensure that the security services are up to date. They actively search for the most recent exploits and system vulnerabilities. Ethical hackers also learn new ways to tamper with electronic devices to maximize efficiency. As a result, they create communities where they can crowdsource their knowledge and develop how people use their devices.

***Grey hat hackers***

As the name implies, they are motivated by white and black hat hacking motivations. They are the ones who use both illegal and legal techniques to exploit or improve a system. Moreover, if a grey hat hacker exploits another person's plan, they usually notify the owner of the exploits and then offer suggestions on improving system security.

***Script Kiddies***

The Script Kiddies are a type of hacker who is an amateur in hacking. They are trying to hack the system using scripts written by other hackers. They are trying to compromise systems, networks, or websites. The goal of hacking is to gain attention from their peers. Script Kids are people who are unaware of the hacking process.

***Red Hat Hackers***

Red Hat Hackers, sometimes called Eagle-Eyed Hackers, are the same types of hackers as white hackers. The red hat hackers intend to thwart the black hat hackers' attack. The distinction between red hat and white hat hackers is in the process of hacking through intention. Red hat hackers are ruthless when dealing with black hat hackers or combating malware. The red hat hackers continue to attack, and the entire system setup may be changed.

***Hacktivist***

These hackers intend to compromise government websites. They pose as activists, which would be known as a hacktivist. A hacktivist can be a single person or a group of anonymous hackers whose goal is to gain access to government websites and networks.

The data collected from accessed government files are used for personal, political, or social gain.

***State/Nation Sponsored Hackers***

The government hires hackers to gather information on other countries. These hackers are referred to as State/Nation sponsored hackers. They use their knowledge to obtain confidential information from other states to prepare well for any potential threats to their country. The sensitive information aids in being on top of every situation and avoiding impending danger. They only document to their respective governments.

***Malicious insiders or Whistle-blowers***

These types of hackers include individuals working in an organization who can expose confidential information. The intent behind the exposure might be a personal grudge against the organization, or the individual might have encountered illegal activities within the organisation. The reason for disclosure defines the intent behind the exposure.

***Hacking process***

The hackers always perform penetration testing processes. The process involves a functional examination of the target system for potential vulnerabilities from poor or incorrect system configuration, known and unknown hardware or software flaws, and operational flaws in process or technical countermeasures. The hacking process has five phases. These are:

- (a) Reconnaissance
- (b) Scanning
- (c) Access (gaining access)

- (d) Maintaining access
- (e) Clearing tracks

### Reconnaissance

Reconnaissance is the first step, also known as data collection and footprinting. This stage takes a long time. During this phase, you observe and collect all of an organisation's network and server information. You will learn everything there so as to know about the organisation. The methods used for reconnaissance include internet searching, social engineering and non-intrusive network scanning. The reconnaissance phase can last days, weeks, or months depending on the target. The main aim of this phase is to learn as much as possible about the possible enemy. Typically, collecting data on the system user, involves the host and network system.

### Scanning

**Scanning:** After gathering all the target organization's information in this phase, the hacker scans the system for security flaws. The hacker will look for defects such as outdated applications, open services, open ports, and network equipment types using scanning tools such as Nmap and web scanners. In the scanning process, typically, they use three stages as follows:

**Port scanning:** Scan the target for information such as live systems, open ports, and system software running on the host during this phase.

**Vulnerability scanning:** During this phase, hackers look for weaknesses in the target that can be exploited. This scan can be performed automatically.

**Network Mapping:** This consists of

creating a network diagram of available data by locating routers, network topology, firewall servers, and host information. This map may be helpful during the hacking process.

### Gain access

It obtains sensitive data by utilising knowledge from the previous phase. The hackers use this information and the network to launch attacks on other targets. The attackers have some control over other devices during this phase. An attacker can gain access to the system by employing various techniques such as brute force attacks.

### Maintaining access

Hackers have many choices for retaining device access, including creating a backdoor. The hacker can use the backdoor to maintain a continuous attack on the network. The hacker can attack the device over which they have gained control without fear of losing access. Creating a backdoor is sometimes inconvenient since it increases the possibility of a hacker being discovered. The backdoor extends the scope of the IDS (intrusion detection system). Using the backdoor, a hacker can gain access to the system at any time in the future.

### Clearing tracks

Ethical hackers will never leave a trail of their hacking activities. As a result, they must delete all documents and logs information relevant to the attack. The primary goal of the clearing tracks phase is to remove all traces that nobody can find.

### Hacking tools (hardware and software)

Both ethical and criminal hackers have access to an abundance of hacking

tools that can be used to either attack or protect a particular system. These tools can be crowd-sourced from the internet through forums and other online hubs dedicated to hackers. Below are popular hacking tools used today.

### ***Angry IP Scanner (ipscan)***

This tool commonly referred to as ipscan by experienced hackers, tracks computers using their IP addresses and snoop for port numbers to look for gateways that will lead them directly into a targeted computer system. This tool is also commonly used by system engineers and administrators to scan for potential vulnerabilities in their authorised infrastructures. These are examples of open-source software that seems to be freely available.

### ***Kali Linux***

This Linux Operating System is one of the favourites of hackers launched in 2015 due to its abundance of features. This security-centred toolkit allows you to run it right from a CD or through a USB without any installation. It contains most of the interfaces you need for hacking, including creating fake networks, spoofing messages, and even cracking Wi-Fi passwords.

### ***Cain & Abel***

Cain & Abel is one of the most effective hacking toolkits for Microsoft operating systems. This tool allows you to recover wireless network passwords, and user account passwords, and crack passwords using a few brute force methods. It could also be used to record VoIP conversations.

### ***Burp Suite***

When mapping out vulnerabilities

on a website, Burp Suite is one of the essential tools you can use. This tool allows you to examine every cookie on a website and initiate connections within website applications.

### ***Ettercap***

This tool helps launch man-in-the-middle attacks, designed to make two different systems believe they are communicating. At the same time, a hacker secretly relies on a different message from the other. It helps to manipulate or steal transactions or data transfers between systems and listen in on conversations.

### ***Metasploit***

Metasploit is well-known among hackers because it is an effective tool for identifying potential security issues and validating system vulnerability mitigations. It has been one of the best cryptography tools for hackers since it effectively masks the identities and locations of an attack.

### ***Wireshark and Aircraft-ng***

These tools detect wireless connections and hack user IDs and passwords on a Wi-Fi network. Wireshark is a packet sniffer, and Aircraft-ng is a packet capturing suite that allows monitoring of Wi-Fi security with various other tools.

**Note:** Brute force is one of the most effective techniques the attacker and crackers use to crack login details such as passwords and utilises the dictionary attack. While most hackers genuinely think that brute force tactics take too long to crack a password, John the Ripper is one of the more efficient tools for recovering encrypted passwords.

## Penetration testing

Penetration testing entails simulating an attack on an organization's information security arrangements by employing various manual and automated techniques. It should be carried out by a qualified and independent penetration testing expert, a security tester with integrity. Penetration testing focuses on exploiting known vulnerabilities, but it should also use the tester's expertise to identify unknown weaknesses in an organization's security arrangements.

A penetration test is typically an examination of IT infrastructure, networks, and business applications to identify attack vectors, vulnerabilities, and control weak points. There are two common forms of penetration testing as follows:

*Application penetration testing* (web applications): This identifies technical weak points in a network or in the web application.

*Infrastructure penetration testing*: This looks for security flaws in servers, firewalls, and other hardware.

## Penetration testing countermeasures

Penetration test countermeasures should be done. It depends on the threats that harm the computer's system or the surrounding area of the network, like Trojan horse attacks, ARP poisoning attacks, wireless network attacks, and footprinting attacks.

## Trojan countermeasures

To eliminate attacks from Trojans, different rules should be followed such as:

- (a) Refraining from opening suspicious email attachments, blocking unused ports, and monitoring network traffic.

(b) Avoid downloads from unknown sources, install up-to-date security software and antivirus, and scan removable media before completing the integrity, enabling auditing, and installing intrusion detection software.

## ARP poisoning attacks

### countermeasures

To eliminate ARP poisoning attack, the following methods should be done:

- (a) Using dynamic ARP inspection in conjunction with DHCP snooping, ARP-to-MAC bindings can be used to track DHCP transactions
- (b) Using dynamic ARP inspection to defend against MAC spoofing attacks

## Wireless network attacks

### countermeasures

To secure wireless networks from ARP attacks, the following measures should be taken:

- (a) scanning access points;
- (b) changing default parameters;
- (c) deploying wireless IPS, and
- (d) selecting solid passwords.

All the measures above can be done by disabling remote wireless device login and configuring WPA2 with AES for data security, scanning RF, filtering MAC, disabling SSID broadcast, blocking rogue access points, and implementing strong authentication.

## Footprinting Countermeasures

The different countermeasures that should be used against footprinting operations are:

- (a) Providing training and awareness of footprinting,

impact, methodologies, and countermeasures to an organization and its employees.

- (b) Employees in an organization should be restricted from using the corporate network to access social networking sites.
- (c) Devices and servers should be set up to prevent data leakage.



### Activity 5.10:

Carry out the penetration test using the following procedure and produce a test report.

1. Footprinting and reconnaissance
2. Scanning (and vulnerability assessment)
3. Enumeration
4. Attacks or hacks
5. Security measures
6. Reporting

### Exercise 5.4

**Scenario:** Angela is a hacker committed to breaking into the TTCL Company. Eliza uses various passive reconnaissance techniques and gathers extensive information about the company. Angela discovers what model routers are used from network administrator questions/comments in user groups. Again, she finds a complete list of the IT staff and their phone numbers from a personnel directory on the company website. Also, she could find out what services were running using a port scan.

From this scenario, consider the following questions:

1. What reasonable steps could the company have taken to prevent Angela from finding out about company hardware, like router models?
2. What steps should the company take to prevent or reduce the efficacy of port scans?
3. Hashimu is a network administrator for the ATCL Company. Hashim has been passed over for promotion three times. He is pretty vocal in his dissatisfaction with this situation. He begins to express negative opinions about the organization in general. Eventually, Hashimu quits and starts his own consulting business. Six months after Hashimu's departure, it was discovered that a competitor had suddenly duplicated a good deal of the ATCL Company's research. Executives at ATCL suspect that Hashimu has done some consulting work for this competitor and may have passed on sensitive data. However, since Hashimu left, his computer has been formatted and reassigned to another person in the interim. ATCL has no evidence that Hashimu did anything wrong. What steps might have been taken to detect Hashimu's alleged industrial espionage? What steps might have been taken to prevent his perpetrating such an offence?
4. Discuss the countermeasure of hacking via footprint attack.

## Security in cloud computing and the Internet of things (IoT)

### Security in cloud computing

Cloud computing is a network access model that allows for convenient, on-demand network access to a shared pool of configurable resources; for example, networks, servers, storage, applications, and services. Cloud computing service models are detailed and described in chapter seven.

Historically, data centre computing models were based on a client-server model architecture. The design or model relies heavily on a three-tier design process that includes access, distribution, and core switches, connecting relatively few clients and meeting few clients' needs compared to today's cloud services models. In most cases, each server was dedicated to a single or limited set of applications and was assigned IP addresses and media access control addresses.

Today's data centre, which provides cloud services, is anything but respectable, as it is overflowing with activities and services that distinguish it from its traditional cousin. For example, its benefits are now available on-demand, by the minute or the hour; it is elastic in the manner that users can have as much or as little of a service as they want at any given time. The service is fully managed by the provider, so the consumer requires nothing other than a personal computer and Internet access.

The mechanism for addressing security and privacy concerns in cloud computing

The difficulties of cloud computing in terms of data security pave cloud computing to emphasise multiple technologies such as networks, databases, operating systems, virtualization, resource scheduling, transaction management, load balancing, concurrency control, and memory management. The result of any company using cloud computing is increasing threats in clouding. For example, adapting the network that connects the plans in a cloud must be secure.

Furthermore, the cloud computing virtual machine paradigm increases the number of security concerns. For instance, the mapping of virtual machines to physical machines must be done securely. Data security also entails encrypting and ensuring that proper data sharing policies are in place. Furthermore, resource allocation and memory management algorithms must be secure. Finally, data mining methods could help detect malware in the cloud.

Then, areas of cloud computing involving the security of data at rest, security of data in transit, authentication of users/applications/processes, robust separation of information belonging to different customers, and cloud legal and regulatory issues should be considered.

Encryption is also the best option for protecting data in transit. Furthermore, authentication and integrity protection mechanisms ensure that data only travels where the customer wants it and is not altered in transit. Strong authentication is a must-have for any cloud deployment. The primary basis for access control is

user authentication. Because the cloud and its data are accessible to anyone via the Internet, authentication and encryption are more critical than ever in the public cloud.

Separation of a cloud provider's users (who may be competing companies or even hackers) is one of the more obvious cloud concerns to avoid accidental access to classified data. A cloud provider typically uses virtual machines (VMs) and hypervisors to separate customers. Currently, technologies available can significantly improve the security of VMs and virtual network separation. Furthermore, the trusted platform module (TPM) can perform hardware-based confirmation of hypervisor and VM integrity, helping to ensure good network separation and security.

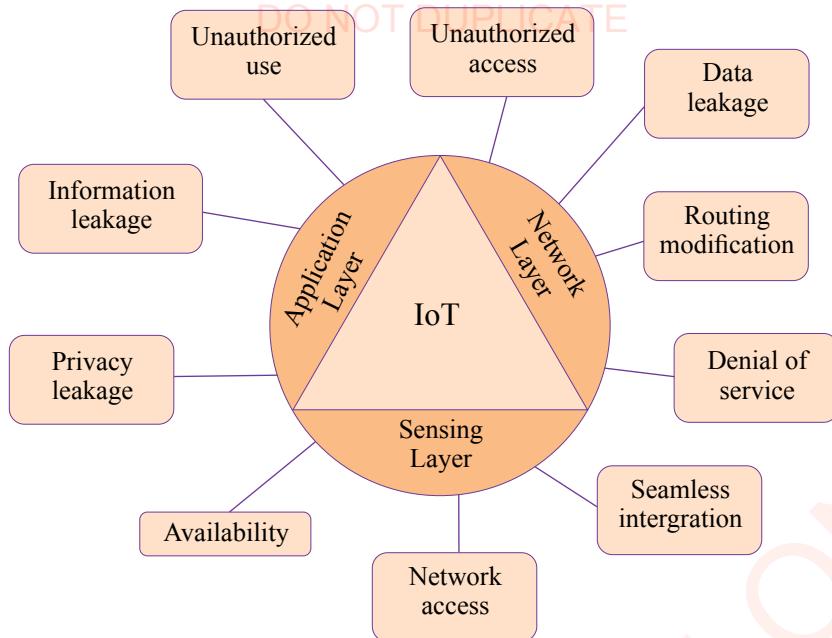
Finally, legal and regulatory issues are critical in cloud computing because they have security implications. To ensure that a cloud provider has solid policies and practices to address legal and regulatory issues, each customer must have its legal and regulatory experts inspect its policies and procedures to ensure their sufficiency. Information security and transmission, compliance, auditing, data retention and destruction, and legal discovery are all issues to consider. Trusted storage and trusted platform module access techniques can play a crucial role in limiting access to sensitive and vital data in data retention and deletion.

## Security in the Internet of Things (IoT)

The Internet of Things (IoT) is defined as sophisticated interaction between the physical and digital world using sensors and actuators that support computation and network capabilities in any conceivable object located anywhere. Security in the IoT requires special treatment because most Internet security measures need high-processing machines amid the limited-processing capabilities of sensors and actuators used in most IoT systems. Due to the limited resource capability, the security of IoT considers three layering perspectives of sensing, network, and application that map functions in support of IoT systems.

### Security threats in IoT

As depicted in Figure 5.32, IoT systems have different security threats that need different measures depending on the threatened function. For instance, the common security threat to the sensing layer of IoT systems includes the unavailability of devices, network access, and seamless integration of the device. Common network layer threats include routing modification, congestion that leads to denial of service, unauthorized access to the network link, and data leakage. The common threats to the application layer include unauthorized use of collected data, information leakage



**Figure 5.32:** Common security threats of IoT

### Security requirements of IoT

In addressing measures for IoT security, one needs to consider a wide range of requirements for sensing devices, communication between the devices and services, and applications of the services. The following subsections discuss the IoT security requirements of the three layers of the IoT system.

#### Security requirements of IoT sensing layer

The main security requirements for the IoT sensing layer should set a secure environment for IoT sensing devices. Security measures in this layer must be scalable to guarantee easy integration of sensing devices. It should offer strong authorization measures against malicious

network access, guarantee a longer lifespan from ambient temperature/humidity/dust, and availability of the sensing device.

#### Security requirements of IoT network layer

The security requirements for the IoT network layer must be robust and self-healing to ensure communication channels support mobility/dynamic without allowing route modification and/or data leakage. The security must also fortify strong authorization across the network and federation of administration domains for managing networks in different authorities and multiple domains to reduce denial of service attacks as well as support decentralized service management.

## Security requirements of IoT NOT DURING EXAM

### application layer

The security requirements for the IoT application layer must guarantee end-to-end security, incremental deployment and privacy to ensure privacy and information leakage as well as authorise the usage of information analyzed from collected data.

who have misplaced or neglected their badge installation of locks and a camera system. Therefore, an organisation has to employ security personnel; this authentication method may include electronic images for the guard to examine or a call to another employee to vouch for the individual. Some businesses solely rely on the employee's signature in the appropriate register.

## Physical security

### Concepts of physical security

Physical security protects personnel, hardware, software, networks, and data from physical actions and events that could cause profound loss or damage to an organization or institution. To enhance physical security the organisation they use one or more options such as installing camera, locks and deploying security guards.

### Categories of physical threats

When someone attempts to steal someone else's computer resources or information, this is referred to as theft. For example, when a fire occurs or there is heavy rainfall leading to flooding causes damage to computer resources. This lead to losing personal or organisation's important information.

### Physical access control systems

Physical control involves security devices such as installation (CCTV camera), and biometric fingerprint. Employees with cameras watching their restroom trips or who must wear badges to enter a facility will need time to adjust to the new circumstances. Suppose badges are to be provided to employees; in that case, the organization must also implement a system for tracking the procedure for dealing with employees



### Activity 5.11:

Search from libraries and the Internet specifically on the concept of physical access control. Write a brief description of physical access controls using the following heading:

Meaning, types, importance and methods of enforcing physical access control in security.

### Physical security controls

Physical access controls are an essential layer of preventive or detective controls that supplement or complement other forms of control in mitigating the risk of inappropriate or improper access and information modification.

### Types of physical security controls

There are various types of physical control such as deterrent, detective, and preventive.

#### Deterrent

Deterrent controls are intended to avoid anyone attempting to sidestep security controls, whether the intruder is from either external or internal. A variety of measures could be considered deterrents, including several overlapping measures.

Examples of deterrent measures include the installation of CCTV cameras, biometric fingerprint locks and sensors.

### Detective

Detective controls are used to detect and report on potentially harmful events. Burglar alarms and physical intrusion detection systems are classic examples of detective controls. Systems are used to monitor for indicators of unauthorized activity, such as doors opening, glass being broken, movement, and temperature changes. They are often set up to watch for undesirable environmental circumstances, such as flooding, smoke and fire, and power outages.

### Preventive

Preventive controls are used to keep unapproved entities from trying to enter our physical security. A primary mechanical lock is a good demonstration of preventative protection. Locks are nearly ubiquitous for preventing illegal access to available facilities, such as enterprises, houses, and other places.

### Corrective controls

This includes any measures taken to repair damage or restore resources and capabilities to their prior state following an unauthorized or unwanted activity. Examples of technical corrective controls include patching a system, quarantining a virus, terminating a process, or rebooting a system. Implementing an incident response plan is an example of an administrative corrective control.

### Protecting people

People are the most important resource in any organisation that means to be protected because they have some information in their brain or in mind.

### Importance of protecting people

The primary goal of physical security is to protect the individual's business and those close to us. At the same time, you put security measures and backup systems in place to ensure that our facilities, equipment, and data remain in functional condition. If someone loses the people when depending on working with the equipment and data, you have a rather complex problem.

### Physical concerns for people

Any adverse physical conditions may be problematic or harmful to the integrity of our data, regardless of the type of digital downloads on which it is stored.

### Safety

Protection of people falls above any other concern and must be prioritized above saving equipment or data, even when such actions will directly cause such items to be damaged. You might find an example of this in the fire suppression systems in use in some data centres. In many cases, the chemicals, gases, or liquids used to extinguish fires in such environments are very harmful to people and, in some cases, can kill them if used in such an environment, as halogen-based fire suppression systems. For this reason, fire suppression systems are often equipped with a safety override that can prevent them from being deployed if there are people in the area. If we were to avoid the suppression system from extinguishing the fire because we knew a person was still in the data centre, we might lose all the equipment in the data centre and potentially data that we could not replace. It would still be the correct choice to make with human life at stake.

## **Evacuation**

Evacuation is one of the best methods to keep our people safe. In almost any dangerous situation, an orderly evacuation away from the source of danger is the best thing you can do. When you make a thorough plan for an evacuation, you ask yourself four basic principle questions of where, how, who, and practice.

### **Where**

Where we will be evacuating is an essential piece of information to consider in advance, whether we are evacuating a commercial building or a residence. We need to get everyone to a rallying point to ensure a safe distance and that you can account for everyone. If we do not do this orderly and in consistent fashion, you may have various issues. In commercial buildings, evacuation meeting places are often marked with signs and on evacuation maps.

### **How**

When planning such routes, you should consider where the nearest exit from a given area can be reached and alternate routes if some courses are impassable in an emergency. Also, avoid using dangerous sites or unusable exits in emergencies, such as elevators or areas that might be blocked by automatically closing fire doors.

### **Who**

The most vital portion of the evacuation is to ensure that you get everyone out of the building and that we can account for everyone at the evacuation meeting place. This process typically requires at least two people to be responsible for any given group of people: one person to ensure that everyone is liable to be safe.

### **Practice**

Particularly in extensive facilities, a complete evacuation can be a complicated prospect. If you do not evacuate quickly and correctly, many lives may be lost in a genuine emergency. As an unfortunate attestation to this, you can look at an example of attacking the 1998 US embassy in Dar es salaam.

### **Administrative control**

Administrative systems are typically based on a set of rules. They can be policies, procedures, guidelines, regulations, laws, or other similar bodies, and they can be implemented at any level, from informal company policies to legal provisions.

Businesses put several standard practices to protect our people and our interests in general. The background check is one of the most common. When candidates have progressed far enough through the hiring process, they are likely to be hired. Often, the hiring company will conduct a background check. Accu Screen and LexisNexis are significant firms that work such background checks worldwide. Depending on the location, such investigations will typically include criminal history statements, past employment verification, proof of schooling, credit checks, drug testing, and other items.

Examples entail drug tests administrated by certain employers. Also, visit any checks expected at the start of employment and are repeated similarly.

Whether or not such verification is performed frequently depends on the specific employer in question, and some companies may not conduct them at all.



### Activity 5.12: Creating and designing authentication credentials

Suppose you have been hired by one of the organisations dealing with national security issues which keeps top secret and classified information, it dedicates every employee with their own room and shares one; how can you design and create authentication and credentials that ensure unauthorised users when enters their dedicated room with sensitive data is denied.

#### Exercises: 5.5

1. Mention and describe three main categories in which we are typically concerned with physical security.
2. Give three examples of physical control that constitute a deterrent.
3. Describe the categories of physical threats.
4. By using the internet facility or other resources, write about the various IoT applications and then, describe how you can help your community address various social and economic problems while engaging in their daily activities.
5. What is residual data, and why is it a concern when protecting security?

### Software piracy and copyright

Software theft and piracy are rapidly ever-increasing problems in the present-day software industry. Due to the evolution of software development and the Internet, software piracy has become the main worry for numerous software

companies. Software companies are threatened with very high losses caused by software piracy.

#### Fundamentals of software piracy and copyright

##### Software piracy

Software piracy is the illegal copy and use of software in a way other than that is formally documented by exclusive rights of the developer in the form of an individual or organization as described in the relevant sale agreement, the license.

Moreover, software piracy is the illegal copying, installation, use, distribution, or sale of software in any way other than that is expressed in the license agreement. Piracy of software, hard-disk loading, soft lifting, counterfeit goods, rental software, and bulletin board piracy can be performed by the end users as well as dealers.

#### Legal issues concerning software piracy

##### *Types of software piracy*

Software may get rid of different computer users in several ways. Some of these ways are as follows:

##### *Internet piracy*

Internet piracy is now one of the most straightforward techniques to achieve pirated software. In some ways, several websites make software available for free download. Many computer users download software from the Internet, which simplifies their lives by trying to eliminate the need to visit stores or send out CD-ROM or floppy disc copies of software. These techniques have made our lives easier, but they have also increased internet piracy, which takes multiple kinds, like downloading

or uploading illegal copies of software over the World Wide Web.

### **Hard-disk loading**

Hard-disk loading occurs when the individual or business sells computer systems preinstalled with illegal software updates on hard discs to entice customers to purchase their products because this type of activity is intended.

### **Software counterfeiting**

The illegal duplication and sale of copyrighted software to appear authentic are known as software counterfeiting. Counterfeit software includes the original legitimate computer's accompanying manuals and is typically sold below the legitimate software's retail price. It is a type of software piracy, and numerous organizations are working hard to stop it. Though counterfeit software is inexpensive and widely available, it is dangerous because it may contain ransomware.

### **Unauthorized use of academic software**

Many software companies sell academic versions of their products to public schools, universities, and other educational institutions. The software labelled only for academic or educational purposes cannot be used for commercial or other for-profit purposes. Using educational software for individual use in violation of the software licence creates software piracy. It harms both the software publisher and the institution entitled to receive the software.

### **Renting**

Renting comprises someone renting out a copy of the software for business or for short-term use, without the copyright holder's permission. In this

type of piracy, the software is rented to individual computers and then returned to the renter. Because of the nature of its distribution, this method of piracy is not as standard as other forms of piracy, but it still occurs. Renting unauthorised access copies of software has always been illegal.

### **Ways to reduce software piracy**

Enforcing strong laws, introducing penalties, creating awareness and selling software at affordable prices can minimise software piracy.

Software companies are required to implement anti-piracy protection mechanisms on their software-based products.

**Enforcing strong laws:** The government and software associations should enforce strong laws/ acts and creates awareness among users. The users should understand that, using pirated software carries high penalties under copyright law for users caught in the act. They can be penalised and sentenced to jail or both punishments.

**Product key:** The anti-piracy system of using the product key is the most reliable. It uses a unique combination of letters and numbers to differentiate copies of the software. A product key ensures that only one user can use the software per licence or purchase.

**Legal protection:** Companies should ensure that their software is legally protected by implementing user agreements. This practice will inform consumers that making unauthorized copies is against the law. The user agreement will help prevent people from unknowingly breaking piracy laws.

**Watermarking:** Watermarks and company logos, or names placed and stamped on software interfaces, indicate that products are legitimately obtained and are not illegal copies by the end user.

**Crack-proofing:** Some software programs have built-in protocols that cause the program to shut down and stop working if the source code is attempted with, modified, or cracked.

*Crack-proofing, called Tamper-proofing, prevents people from pirating the software by manipulating the program's code.*

### Software copyright

#### Copyright

The sole right that an intellectual property owner has is called copyright. Copyright is used to protect various original works such as paintings, photographs, computer programs, video games, and actual databases for the programmer and software vendors. These rights were initially established by courts drawing a parallel comparison between the source code and the executable program itself and blueprints for architectural structures and the structure itself. Examples of copyright are novel poems, movies, lyrics to a song, musical compositions in the form of sheet music, sound recording and painting.

#### Importance of software copyright

Copyright protection enables businesses to prevent copying, limit competition, and charge licensing fees and monopoly prices for the produced products. Apart from that, software copyright increases national per capita income as an individual and nation.

Nevertheless, every person's intellectual skill is protected and built-up ownership of property legally without interfering, hence, generating individual money when selling their work. For example, the author produces many books and sells them in various schools.

#### Types of software copyright

*Copyright Software Licensing:* This type of user agreement protects the intellectual property of a software developer who created an application, its source code, and its object code.

*Copyright law:* This area of law covers intellectual property – when the idea is the product. For example, the copyright for a book is about the words, not the paper, ink, and binding.

#### Legal implications upon abuse of a copyrighted material

Various schools of thought have addressed the challenges facing the copyright protection of works in digital form (Mwim & Pistorius 2017:1029–1030). Legal scholars' philosophical approaches highlight the essential characteristics of the digital copyright paradigm. These schools of thought range from abolishing copyright protection of works to a moderate approach that embraces the public-private balance of rights. Extreme minimalists argue that copyright law is dead.

#### Exercise 5.6

- Rose Joseph gave Juma Hassan some software to copy onto his computer, and now he realizes he is guilty of piracy. What would he do?
- Mwanampaka schools purchased a single license of a piece of Windows 10 Operating system software. They gave an IT technician the task of evaluating whether it could be loaded onto several machines. As a computer science student, advise the IT technician to perform the task.

3. Many companies use resellers and third-party distribution channels. How can you tell if a vendor has the legitimate right to sell another company's software?
4. What are the risks associated with using pirated software?

Moreover, institutions' quality also depends on intellectual property, which strongly bears competitiveness and growth. It influences investment decisions and the organization of production. It plays a central role in how societies distribute the benefits and bear the costs of development strategies and policies. For example, owners of the land, corporate shares, or intellectual property are unwilling to invest in the improvement and keep up their property if their rights as owners are insecure.

Finally, intellectual property promotes innovation. The different forms of intellectual property, such as patents, copyright and trademark, were similar to traditional legal rights viewed as state favours of moral rights. However, IP rights are an essential economic mechanism, an 'intellectual currency' of sorts.

### Legal issues on intellectual properties

They are focused on the use and ownership of computer software which is essentially a question of power relations between persons today and for many years into the future;

1. the forms of legal protection provided are both contingent and far from coherent in many dimensions; in other words, such forms of protection are not inevitable and could be changed, indeed quite dramatically changed, through reforming the structures and content of legal regulation;
2. the granting of intellectual property to computer software is a form of legal subsidization to a particular industry and technology; and

## Intellectual property

Intellectual property refers to any creation of human mind that is protected by the law from unauthorised use.

Intellectual property is of the several distinct types of legal monopolies over creations of the mind, both artistic and commercial, and the corresponding fields of law. Under intellectual property law, owners are granted certain exclusive rights to a variety of intangible assets, such as musical, literary, and artistic works; discoveries and inventions; and words, phrases, symbols, and designs.

Copyright, patent, and trademark, and trade secrets are protected by both national and international laws.

## Importance of intellectual properties

Intellectual properties usually generate economic activity and jobs in other sectors. Domestic companies in developing countries also rely on the trademark system to protect their brands at home and abroad. As with patents, trademarks are on the rise in developing countries. Domestic companies constitute a substantial part of the users of these trademark systems. Compared to developed countries, there is a high rate of use of the trademark system in developing countries relative to the Gross Domestic Product (GDP).

3. the intellectual property regimes that protect computer software have directly impacted the ownership and user regimes that have been established; the alternatives to proprietary, open-source, and free software have been a philosophical and practical response to the current legal issue.

### Categories of software licenses

There are two main categories of software licenses: Open source and proprietary (non-open-source) software.

#### Proprietary software

Proprietary software is owned as private property by a company (or occasionally by an individual software developer). Various intellectual property laws and regimes protect its ‘private proprieties’.

**Free Software/Open Source:** These are highly tolerant pacts that allow users to freely modify, use, share, and reuse a software product’s source code. Free Software/Open Source (FOSS) licences give users a great deal of control over using the software. According to this alternative approach, all users have “the right to use the software they want, for whatever they want, on as many computers as they want, in any technically appropriate situation.”.

Nowadays, software are available to meet their standards. Of course, this includes improving it, fixing bugs, expanding its functionality, and researching how it works.

Software allows users to distribute the same software to other users so that they can use it as needed. This redistribution can be done for free or at an expense that is not predetermined.

### Difference between open source and proprietary software

The main feature that makes proprietary software different from open source is that, in open source, the user has the freedom to use the software, own it free and redistribute it to other users rather than proprietary software which is intended for selling for earning money.

### Exercise 5.7

1. Identify and distinguish between open and proprietary software
2. Operating system software like Linux is perhaps the most widely known example of open-source software. What are some others that people may not be aware of?
3. Kisiwa Ndui Secondary purchased a brand-new computer, but they did not have an operating system software system, and they had no extra money to buy Windows OS. From this scenario, as an expert in IT, what is the solution to help Kisiwa Ndui without breaching the copyright laws and proprietary on free of charge?

### Ethical and Legal Issues

#### Ethics

Ethics is a branch of philosophy that deals with right and wrong. It is a system of principles and rules of conduct recognized and accepted by a specific group or culture. Ethics is a field of study concerned with distinguishing right from wrong and sound from bad in computing.

## Legal issue

The legal issue in a computer system is the area of an individual's right to privacy versus the greater good of a larger entity such as a company or a society.

For example, tracking how employees use computers, crowd surveillance, managing customer profiles, and monitoring a person's travel with a passport. The legal issue is ensuring the security of stored and transmitted information.

The laws currently governing commercial transactions, data privacy, and intellectual property were primarily developed for a time when telegraphs, typewriters, and mimeographs were the commonly used office technologies, and business was conducted with paper documents sent by mail. Computers, electronic networks, and information systems are now used to process, store, and transmit digital data in most commercial fields. As the spread and use of information technologies in the business world have quickened, the failure of current laws to meet the needs of a digital, information-based society has become apparent.

## Importance of ethical and legal issues

Firstly, electronic businesses replace conventional paper documents with standardized computer forms. The need arises to secure the transactions and establish means to authenticate and provide nonrepudiation services for electronic commerce to verify the authenticity and certify that the transaction was made. The absence of a signed paper document on which any unauthorised changes could be detected,

a substitute for the signature and a means to prevent, avoid, or minimize the chance that the electronic document has been altered must be developed.

Moreover, the protection of privacy in data and international policies pave the way for the rapid development of networks and information processing. Today, the computer allows large quantities of personal information to be quickly acquired, exchanged, stored and coordinated. As a result, the market for computer-matched personal data has expanded rapidly, and the private-sector information industry has grown around the demand for such data.

Lastly, intellectual property is protected in the administration of digital libraries. The availability of protected intellectual property in networked information collections, such as digital libraries and other digital information banks, is straining the traditional methods of protection and payment for the use of intellectual property. Technologies developed for securing information hold promises to monitor the use of protected information and provide a means for collecting and compensating intellectual property owners.

## Ethical issues and their legal implications

Social media raises legal and ethical issues that must be carefully assessed and considered for their risks before creating and disseminating information. The legal challenges faced involving social media user rights and copyright laws. In contrast, activities conducted by social media such as the Jamii Forum, Facebook, WhatsApp, and Twitter in

our countries threaten individual life through interactions made by social media. Current laws fail to address social media directly, and discussions and conflicts regarding their legal standing are ongoing in the courts. As legal battles over social media continue, laws and legal implications are subject to change. (See Tanzania Cyber Crime Act 2015). In addition, intellectual property rights pervade the discussion on legal dilemmas that researchers, archivists, librarians, and others confront when harvesting social media data. Digital platforms have become increasingly complex, and social media use has escalated, creating new avenues of research data. Organizations must keep pace with research demand “in a rapidly changing environment characterized by new distribution mechanisms, expanding copyright monopolies, ever-greater technology dependencies, and changing user expectations.” Hence current intellectual property regulations are not easily applied since much of copyright law remains unchanged and outdated in an increasingly digital age.



**Activity 5.13** Consider the Tanzania Cybercrime Act 2015. Discuss the following cases regarding the act, the offence and the consequences

1. Someone pirates the CD and distributes it by selling it without copyright.
2. Someone publishes someone's photo on social media without his/her consent.

### Exercise 5.8

1. Mention and describe types of Software Piracy
2. Distinguish between ethical and legal issues.
3. Consider the following Scenario: Fatuma has been employed by TBC company working as an accountant in the Accounting Department. She used to change figures, or double amount of salary of some workers, to make her own profit. From this scenario, which section and subsection of the Tanzania Cyber Crime Act 2015 were violated?

### Chapter summary

1. This chapter discusses computer security as the protection of computerized information systems to prevent illegal access, use, transmission, disruption, tampering, or destruction of information system resources. Such protection enables users to safeguard access to information system resources like hardware, software, data, and mobile phones.
2. Moreover, the techniques ensure assurance of communication both to users and devices. Access control was established by the tools used to prevent a network, such as firewalls, IPS/IDS and VPN, from being installed and checking if an attacker was hacking the system or not by using Kali Linux software tools for advanced Penetration tests. Kali Linux is an open-source,

Debian-based Linux distribution geared toward various information security tasks, such as Penetration Testing, Security Research, Computer Forensics and Reverse Engineering.

- Lastly, the advancement of technology brought ethical hacking into a business in society; consequently, cybercrimes, terrorism crimes and money fraud robbery activities give a chance to nourish. Increasing crimes in society lead to existing software piracy and copyright that make all misbehaviours and criminal offences should be dealt with according to its laws.

### Revision exercise

- Discuss in detail the risk assessment process.
- A friend sends an electronic e-greeting card to your work email. You need to click on the attachment to see the card. A while back, the IT help desk received several complaints that one employee's computer sent out Viagra spam. They checked it out, and the reports were accurate. A hacker had installed a program on the computer that automatically sent out tons of spam emails without the computer owner's knowledge. How do you think the hacker got into the computer to set this up?
- The mouse on your computer screen starts to move around and click on things on your desktop. What should you do?
- Distinguish between a Copyright and a Trademark.
- What is the defining difference between computer security and information security?
- How informed do you keep yourself on network security-related news, and how often do you check out these stories? Where do you get your security news?
- Do you think, for security reasons, everything that happens on the internet should be analysed by public security services?
- Elaborate on the following five cybersecurity risks in the banking industry: (a) Unencrypted data, (b) Malware, (c) Third-party services that are not secure, (d) Data that has been manipulated (e) Spoofing.
- Define the term hijacking as it relates to cybersecurity.
- Evaluate the weakness and the strengths of the Tanzania Cybercrime Act 2015
- You discover an active problem in your organization's network, but it's out of your sphere of influence. There's no doubt that you can fix it, though, so what do you do?
- What type of physical access control might you put in place to block access to a vehicle?

# Information technology career and environment

## Introduction

Advances in Information Technology (IT) can never be discussed without associating them with users (people) and their environments. Users' knowledge of the benefits of IT and their ability to apply that knowledge can lead to a successful personal career. IT facilitates organisations to achieve their objectives and helps society to acquire a superior quality of life. Hence, communities, businesses, and industries worldwide reap huge benefits from IT if it is deployed intelligently and carefully. Therefore, information systems must also be developed to be accepted and work well and safely within their environments based on the intended business and country's goals and strategies. In this chapter, you will learn more about IT career opportunities and the IT environment. The competence gained will enable you to effectively manage businesses or guide you in making the right IT career choice.



### Think about the following:

Which IT professional and personal competencies are you interested in, and how are you expecting to develop them?

How can you use the career perspective as a mirror to reflect on your future IT career?

What is the future of IT careers in the jobs market, and how IT helps and provides support to other professionals?

Imagine how the world of work could be without Information Technologies.

## IT career opportunities

Information Technology is one of the most competitive professions in the market. Professions like IT Security, Artificial Intelligence (AI), Cybercrime,

and software engineering are some of such competitive IT occupations. Since most companies and organisations have automated their core functions, IT skills have become inevitable in the different dimensions of our life. Regardless of what industry you anticipate joining, knowledge of information technology can help someone stand out as a competitive candidate. As information technology becomes increasingly integrated into daily life, the opportunities available for those with IT skills and competence to operate such IT systems have expanded. Therefore, studying IT may be an excellent choice if a person is interested in being an IT professional.

### Roles of various IT career opportunities

There are several career opportunities in the IT industry. Today, the demand for

IT experts is rising because companies, institutions, and governments are revolutionising how they undertake their daily activities. In that regard, there are many areas where a person can work as an IT professional. For example, when you anticipate working as a system administrator, there are several roles, that you are required to perform. The most common career opportunities are such as, Computer operator, Computer technician, computer system analyst (CSA), System administrator, Computer programmer, Network architect/or Network administrator, Cloud computing engineer, Database administrator, IT project manager, Web developer, Information security officer, Business Intelligence Analyst, and Mobile application developer.

### **Computer operator**

According to standardised operating instructions, the computer operator monitors and manages computers and electronic data-processing equipment. Depending on the employer or employing institutions, key roles of the computer operator might include:

- (a) Managing and improving a company's network performance by troubleshooting hardware and software problems
- (b) Entering batch data into the computer for processing
- (c) Logging all information processing activities and exceptions
- (d) Performing backup to reduce the risk of data loss
- (e) Identifying and correcting file and system errors

- (f) Performing data processing operations according to a business production schedule
- (g) Maintaining computer equipment and inventory and organise repairs as needed
- (h) Creating and managing users credentials to ensure the security of the organization.

### **Computer technician**

A computer technician regularly maintains, upgrades, and repairs computers and their accessories. The computer technician is on the front lines of IT and performs the following duties:

- (a) Troubleshooting and solving problems related to hardware and software
- (b) Assembling, upgrading computers and installing software patches and upgrades
- (c) Monitoring servers and networks
- (d) Maintaining computers and peripherals, and managing data security
- (e) Ensuring that computer systems are all optimised and run efficiently
- (f) Addressing users' explanations of computer problems and evaluate their needs
- (g) Guiding users through the steps necessary to resolve their problems and train them to use computer hardware and software properly.

### **Computer Systems Analyst**

A Computer Systems Analyst (CSA) is responsible for analysing and designing hardware and software tools to improve

overall IT function in the organisation. The CSA is also responsible for identifying the organization's needs and problems and then designing a solution for the problem. The roles of CSA include the following:

- (a) Managing the installation, deploying, and testing of new computer systems
- (b) Training end-users and preparing instruction manuals and other essential documentation
- (c) Developing new and upgraded functionality for existing computer systems
- (d) Collaborating with managers and other experts to assess an organisation's IT needs
- (e) Researching and analysing the possible advantages and drawbacks of emerging technologies
- (f) Selecting and configuring the new hardware and software to meet the organisation's needs.

### **System administrator**

A system administrator is essential in organisations that have computerised their organisational functions. A system administrator is responsible for system maintenance, data management, crisis management, and profile management. The following are some responsibilities of a system administrator:

- (a) Install, configure, and troubleshoot software and hardware issues and outages
- (b) Manage the network servers and monitor their performance according to requirements

- (c) Set up users' accounts and ensure security through access controls
- (d) Upgrade the systems with new updates and releases
- (e) Maintain records of organisational IT assets usage
- (f) Train end-users on new emerging technologies
- (g) Build internal technical documentation, manuals, and IT policies

### **Computer programmer**

A computer programmer, also known as a software engineer, is responsible for designing and creating systems and software programs. Some key responsibilities of a computer programmer include the following:

- (a) Systems integration and software development
- (b) Examine users' needs and create computer programs that address those needs
- (c) Train end-users on newly deployed software and systems
- (d) Manage database systems
- (e) Analyse algorithms, write computer code, and modify source code
- (f) Design and test computer structures and programs
- (g) Write system instructions, debug, and maintain operating systems
- (h) Planning and modelling software
- (i) Upgrade the existing programs and repair program errors

## Network architect/ Network administrator

A Network administrator is responsible for analysing, defining, designing, building, and maintaining various data communication systems and networks, including LANs, WANs, and the Internet. The central roles of a Network administrator include the following:

- Analyse the workflow and network security needs and establish the network
- Assess network performance problems and provide solutions
- Network installation and testing of equipment, connections, and firewalls
- Define the policies and procedures related to the network
- Monitor and analyse the network and troubleshoot related problems
- Fine-tune the network to enhance the performance for users
- Implement, monitor, and evaluate security protocols
- Train the end-users on the safe and secure use of the network
- Test, evaluate, install network enhancements periodically and update the network
- Maintain accurate documentation and records of the network

## Cloud computing engineer

A cloud computing engineer is responsible for defining, designing, implementing, and maintaining systems and solutions that rely on cloud systems, platforms, and infrastructure. The three major types of cloud Engineers are

*Solutions architects, Cloud developers, and System operations engineers.* The key responsibilities of a cloud computing engineer may vary depending upon the vertical position. Cloud computing engineer may have many other subcategories of career, including;

- Cloud security engineer
- Systems engineer
- Cloud developer
- Cloud architect
- Network engineer

A Cloud computing engineer is responsible for the following responsibilities:

- Developing and maintaining cloud solutions following best practices
- Designing and developing modular cloud-based systems
- Ensuring efficient functions of data storage and processing functions in the cloud infrastructure
- Responsible for securing cloud infrastructure and its components
- Conducting and managing data analytics in the cloud environment
- Providing technical support for cloud clients and users in the organization

## Database administrator

Nowadays, organisations and businesses produce and collect large amounts of data to run their daily operations. Therefore, the database administrator is needed to store, secure, and organise the company's or organisation's massive data. The roles of a database administrator are as follows:

- (a) To perform administration of systems, databases, server virtualisation and server infrastructure
- (b) To manage security aspects to the assigned systems, database, integrity controls, related records and documents
- (c) To install, upgrade, and maintain software applications and databases
- (d) To undertake daily maintenance, testing, backups, and recovery of systems and databases per the organisation's policy and standards
- (e) To ensure high availability of systems is working efficiently and services
- (f) To monitor standards, procedures, and access methods for the database management system

### IT project manager

The IT project manager is responsible for planning, initiating, and executing complex IT projects and initiatives. IT project manager is accountable to:

- (a) meet with the organisations' stakeholders to help determine the scope and timing of IT projects;
- (b) work with technical teams to identify barriers or issues and determine options for how to address them;
- (c) conduct regular checks to monitor project progress and address unexpected issues;
- (d) create overall IT project work rolling out institutional plans;
- (e) consult technical teams to identify what resources are needed for a project; and

- (f) develop and implement the ICT policy of the organisation.

### Web developer

A web developer is responsible for designing, creating, and maintaining web applications. An individual can choose to work as: a back-end developer (working mainly with the technical development of the website), a front-end developer (responsible for the look and design of a website), and a webmaster (maintain and update websites). It is also possible to perform all these roles depending on the arrangement. A web developer is responsible for:

- (a) conducting meetings with clients and evaluating their website development needs;
- (b) building and testing the web applications and integration with respective databases;
- (c) code writing for the website;
- (d) cooperating with the website stakeholders' team;
- (e) creating user-friendly, operational, and accessible layouts;
- (f) cracking and monitoring the site traffic or other key performance measure or analysis.

### Information security officer

An Information security officer protects and manages the organisation's network and systems against security vulnerabilities and potential fallout. Information Security Officer manages and creates disaster recovery plans, conducts assessments of security threats, develops procedures for resolving potential threats, and keeps an eye on the new threats in the industry. The roles of

IT security officer include the following:

- Monitor the systems for security risks or possible vulnerabilities
- Supervise the organisation-wide implementation and management of safeguard measures such as firewalls and antivirus software
- Prepare reports related to security risks and guide on the prevention measures
- Ensure timely systems testing to identify potential weaknesses and implement solutions
- Monitor network usage to ensure compliance with security policies

### **Business intelligence analyst**

A Business Intelligence (BI) analyst is responsible for retrieving and analysing data within a company or organisation. The roles of BI Analyst are as follows:

- To make consultations with management and the essential stakeholders to define goals
- To conduct research, develop, and implement data-gathering methods
- To provide analysis/analytics and synthesis of data (descriptive, diagnostic, predictive, and prescriptive data analytics)
- To report on research findings and recommend solutions
- To collaborate with co-workers and management in implementing various strategies
- To evaluate the effectiveness of implemented strategies
- To develop and manage BI solutions

### **Mobile application developer**

A mobile application developer is a software developer specialising in mobile technology, such as developing various applications (or apps) for Google's Android, Apple's iOS, and Microsoft's Windows Phone platforms. The typical roles of a mobile application developer include the following:

- To prepare the entire application lifecycle (concept, design, test, release, and support)
- To generate fully functional mobile applications by writing clean code (i.e., coding, testing, debugging, documenting, and monitoring)
- To work together with various departments in the organisation regarding new deployments
- To contribute to the development of project schedules and workflows
- To propose changes and enhancements to software applications
- To prepare the unit and user interface (UI) tests for identifying malfunctions
- To conduct research and propose new mobile products, applications, and protocols
- To stay up to date with new technology trends



### Activity 6.1: Identifying IT opportunities and their respective responsibilities

**Resources:** Computer, Internet connectivity, IT career roles checklist, notebook and pen.

**Procedures:**

1. Visit various organisations' websites in Tanzania.
2. Navigate through the job advertisements pages and identify the IT-related jobs and the roles/responsibilities.
3. For the job opportunity identified in part 6.1 (2), write down the responsibilities assigned.
4. Compare each IT career role and responsibilities assigned to each identified job opportunity and compare them across the organisation
5. Write all similar job roles and responsibilities assigned among all institutions or the organisation you have visited.
6. What do you conclude from part 6.1(1-5)?
7. Prepare a list of IT career roles and their responsibilities in your report.

**Note:** A list of IT careers' roles and responsibilities will be used as a resource in the next activity you need to carry out outside the school.



### Activity 6.2: Exploring IT careers around your community

**Resources:** Computer, Internet connectivity, IT career roles checklist, notebook, and pen.

**Procedures:**

1. Visit at least two websites or institutions of your choice.
2. Record all business operations or functions performed using computerised systems.
3. Identify the roles of each person in the business operations identified in (number 2).
4. Meet with responsible persons in each IT-related department, ask for their roles, and document them.
5. Compare the roles/responsibilities with those you have learnt and those summarised from activity 6.1.

### Exercise 6.1

1. What are the key responsibilities of the following IT specialists?
  - (a) Computer technician
  - (b) Web developer
  - (c) Database administrator
  - (d) Information security specialist
2. Assume you have been employed as a Business intelligence analyst. What would be your key responsibilities?
3. Describe the emerging IT opportunities with their job roles.
4. Suppose you have been allowed to work with the financial firm in the IT department. What IT profession would fit you? State the expected key responsibilities.

## *Relationship of various roles of IT DUE TO career opportunities*

Principally, most of the roles of the IT profession depend on each other. There are various IT careers within the organisation, each having its specific roles to be undertaken by involving various IT experts such as web developers working together with IT security specialists. Nevertheless, each role is essential to the company's overall performance and operations. The IT professional responsibilities do relate to each other in some contexts. Similarly, a web developer sometimes needs an IT security specialist to assess the website's vulnerability before launching. In other words, the two experts can interchange or undertake some responsibilities together.

It can be said that, within the IT industry, the experts depend on each other based on their professional skills. Remember that "you cannot be an expert on everything." You can only do the work related to your skills, and the rest will be done by others whose skills fit into that aspect.



### **Activity 6.3. The relationship between various roles and responsibilities of IT professionals**

#### **Resources:**

Computer or smartphone with Internet connectivity, library resources, IT roles reports, piece of paper and pen.

#### **Procedures:**

1. Visit multiple sources such as the Internet, library, and employers.

2. Identify various roles of IT professionals and explain how they relate to or interact with each other.
3. Use your report and experiences from activity 6.1 to ascertain how these IT carriers roles are related to each other.
4. Share your finding with your fellow in the group to make a relationship among IT careers.
5. Present your findings to the whole class for discussion.
6. Summarise the class discussion critical points on the relationship and interaction among the identified careers.

## **IT and environment**

Technological advancement has had a tremendous impact to our natural environment and on peoples' daily lives. Technology carries both advantages and disadvantages for human beings and the environment. In other words, technology is a two-edged sword, capable of both causing and repairing environmental and people's lives damage. The following are some positive and negative impacts of technology on the environment.

**Air and water pollution:** When dangerous or excessive amounts of gases such as, carbon dioxide, carbon monoxide, sulfur dioxide, nitric oxide, and methane, are produced during manufacturing and use of IT device/equipment, into the earth's atmosphere, they will damage or pollute the air. Air pollution has damaging health effects on people and animals and contributes to global warming. It may

cause rising levels of greenhouse gases in the atmosphere, thus trapping the thermal energy in the earth's atmosphere, causing the rising of global temperature. If not managed properly, the 'end-of-life' of IT products may contaminate water bodies such as the sea, lakes, and rivers, destroying aquatic ecosystems.

*Resource depletion:* This is another negative impact of IT on the environment. Resources (e.g., fossil fuels and minerals) might be consumed quickly to feed the information technology industries before replenishing them.

### Impact of IT on people's daily lives

IT has made life easier, particularly for those who embrace it. People can now utilise the Internet and social media for various purposes, including online trading or business. However, IT has a detrimental impact on peoples' lives. Environmental pollution, for example, can cause human and animal deaths. Countries are currently witnessing the improper management of electronic wastes; thus, putting more risks on the ecosystem.

### Advantages of information technology

The advantages of information technology to people's lives include the following:

- It has made the education process more efficient and productive in teaching and learning.
- In the health system, information technology has improved the production and supply of medicine, medical devices, and e-consultation. The available digital devices in the health sector facilitate medical-related activities such as surgery.

D(c) ITCA has facilitated several services such as building or road construction, weather forecasting, and simulation.

- IT is used to maintain and manage financial and business documents in business, such as Mobile and Simbanking.
- Information technology plays a pivotal role in various countries' political landscapes, especially in electoral campaigns. In this context, technology influences public opinion and mostly increases young people's involvement in political life.
- IT has improved the timely distribution of information through different media and communication in homes and workplaces through social networks and emails.
- It contributes to the national Gross Domestic Product (GDP) growth through increasing revenues due to IT-based economic activities.

### Disadvantages of information technology

The disadvantages of information technology to people's life are as follows:

- Prolonged use of computers, tablets, and cellphones can lead to digital eyes strain, dry eyes, blurred vision, headaches, and shoulder and neck pain. The screen glare, lousy lighting, and improper viewing distance or angle contribute to all these.
- Health-related problems due to the improper dumping of IT 'end-of-life' materials or electronic wastes (e-wastes).

- (c) Reduced face-to-face interaction:  
Nowadays, people mostly prefer online communication rather than face-to-face conversations, so they tend to become more individualistic and introverted.
- (d) Job placement and replacement.  
The use of IT simplify work; the work which can be performed manually by many people can be done by one person using a computer-aided device.



#### Activity 6.4: Identify information technology's impact on the environment and provide ways to reduce its effects.

##### Resources:

The computer with Internet connectivity and notebook.

##### Procedures:

- Open the computer and the web browser of your preference.
- Write the URL of the search engine like google, Aster vista, or dogpile and press enter to open it
- Write the search terms in the search field to search for “the impact of information technology on the environment”. Read and document the impacts.
- Search for, and document the ways to reduce adverse effects
- Then, write your report about the findings using the following guided questions:
  - What are the identified impacts of information technology on the environment?
  - What have you determined as ways to reduce adverse effects on the environment?

#### Exercise 6.2

- Discuss the disadvantages of technology to human lives.
- Explain how youth can make use of technology to employ themselves.
- Explain any five advantages of being an IT expert in the 21<sup>st</sup> century.
- What do you understand by the statement “improper dumping of IT ‘end-of-life’ materials”?
- Discuss any five impacts of IT advancement on the environment.

#### Methods for safe disposal of computer and related hardware

The IT devices keep a lot of information; therefore, when you dispose of, recycle, or donate such devices to others, be careful because you might unintentionally disclose sensitive information, which cyber criminals could exploit. These IT devices may include but are not limited to:

- Electronic gadgets (e.g., computers, tablets, smartphones) which can automatically store and process data.
- Digital media such as digital cameras and media players which can create, store, and play digital content.
- External hardware and peripheral devices (such as printers, monitors, and external hard drives) that contain permanently stored digital characters.
- Electronic gaming consoles, computers, or digital devices that output visual images or video signals or display a video game.

There is a variety of methods for the safe disposal of IT-related materials. Nevertheless, before disposing of IT

## FOR ONLINE USE ONLY

devices, consider backing up the data or cleaning a gadget.

The disposal of electronic waste (or e-waste), which are electronic materials that have reached the end of their useful lives and have to be discarded; can be done in the following ways:

- Donation to a nonprofit organisation
- Recycling electronic gadgets, a strategy that may save natural resources
- Choosing a reliable and certified vendor or company responsible for disposing of outdated electronic materials
- Selling your older hardware is another safe method of disposal
- Refurbishment of IT devices, an action that may extend their lifetime

Therefore, before disposing of computers and related devices, make sure that you;

- backup significant files and information,
- erase all data in your hard drive, and
- dispose of your IT devices.

### **Health hazards associated with the use of IT**

IT-related hazards are potential health problems caused by the prolonged use of IT tools. The prolonged use of IT tools, such as playing games, browsing the Internet, and instant messaging, can contribute to prolonged health problems for users. Some examples of health hazards associated with the use of IT devices include:

- Back and neck pain
- Eye strain
- Pain in the hands and arms

(d) Carpal Tunnel Syndrome (CTS)

- Wrist pain
- Psychological illness such as arthritis, internet addiction, and cybersex addiction
- Reduced physical activities

### **Precautions on IT-related health hazards**

The IT user must understand all health risks associated with using IT devices. As part of the recommendation, first, you should learn how to use these technologies without experiencing any problems rather than avoid them. Some IT users are unaware of the health-related issues they may face due to operating IT devices. Second, users of computer technologies should immediately receive education on the healthy use of these technologies.

Individuals may suffer serious health problems if the necessary precautions are not taken. Consequently, institutions may suffer a significant decrease in the workforce and financial losses if the required precautions are not accepted. Precautions must be taken to reduce users' IT-use-related health problems.

The following are some of the precautions for IT-related health hazards:

- The use of ergonomic chairs to help avoid pain
- Take a brief break to stretch and walk around
- Use of low-intensity light fixtures or the use of LCD or plasma screen
- The use of screen contrast that can not harm your eyes and a moderate lighting environment



### Activity 6.5: Identify health hazards associated with using IT devices.

Search from the Internet and identify any five health hazards associated with using IT devices. From your search, assess the uses of IT devices among the community members and the impact associated with their use.

### Exercise 6.3

1. List down six (6) IT devices you know.
2. Outline five IT-related hazards due to the prolonged use of the computer.
3. Explain with examples the impact of improper disposal of 'end-of-life' computer hardware on the environment.
4. Discuss the precautions to avoid the impact of prolonged use of IT-related devices in your groups.
5. Describe safe methods of disposing of the computer hardware.
6. What precautions should be taken before disposing of the computer hardware?

### Chapter summary

1. Information Technology is one of the most competitive professions in the market. Professions like IT Security, Artificial intelligence, Cybercrime, and software engineering are some of the competitive IT occupations. Since most companies and organisations

have automated their core functions, IT skills have become inevitable in offices.

2. Studying IT may be an excellent choice if you are interested in becoming a computer operator, IT manager, System analyst, System administrator, Computer programmer, IT trainer, Data clerk, Webmaster, IT laboratory technician, and Maintenance engineer.
3. IT has made life easier, particularly for those who can interact with opportunities like utilising the Internet and social media.
4. Advancement in IT has brought about a tremendous impact on the environment, particularly the negative impacts on our environment and people's daily lives.
5. Prolonged use of computers, tablets, and cellphones can lead to digital eye strain, which can easily be realised with symptoms such as; dry eyes, blurred vision, headaches, and shoulder and neck pain.
6. Technology has a detrimental impact on peoples' lives, particularly the youths trying to cope with a changing world and culture.
7. For several reasons, it is essential to decide on the proper ways of safely disposing of electronic devices.
8. Disposing of computers and IT-related devices should be considered for all unused ones. During disposal, proper procedures need to be followed.

**Revision exercise**

1. Search from the Internet and identify the career opportunity of your choice. Explain the roles and necessary skills required for that career.
2. Using different sources, identify IT career opportunities that are currently highly demanded.
3. The advancements in IT are associated with both advantages and disadvantages. Explain them with vivid examples.
4. With examples, describe how one IT profession relates to each other.
5. List down the health-related problems linked to the use of IT devices.
6. Improper disposal of electronic waste may damage the environment and pose various challenges to human health. Examine five of these challenges.
7. Addiction to IT devices like computers and mobile phones is an emerging challenge for youths. Explain any four measures that can be taken to address these challenges.
8. Study various technologies in Tanzanian society and point out any four adverse effects they bring to peoples' healthy and environments.
9. Unemployment has been a serious global issue for youth, particularly in developing countries like Tanzania. It adversely affects their economic growth. So most youths seek limited office jobs. As an IT expert, how would you advise the unemployed youth to take advantage of IT end-of-life disposed of materials to address the issue of unemployment?
10. As a student in IT subject combination, how can you advise people to realise the role of IT in 21st-century socio-economic advancements?
11. End-of-life IT technological materials litter badly colour our surrounding environments. Suppose one of the Community based organisations dealing with environment and youth affairs requested you to give detailed advice on how these end-of-life technological materials disposed of illegally can be an opportunity to address youths' unemployment; what would be your advice?

## Introduction

Since the end of the 18<sup>th</sup> century, the world has gone through four industrial revolutions. In the first, second, and third industrial revolutions, production was based on steam, electricity, and Information Technology respectively while the fourth industrial revolution is derived by cyberphysical systems, which connect intelligent machine, smart gadget, processes and customers. In this chapter, you will learn about emerging digital technologies that drive the 4IR and their relevance, application, and impact. The competencies developed will enable you to understand and apply emerging technologies and trends driving 4IR in your community and industry.



### Think about the following:

1. Using a smartphone's camera to view buildings along a street and then be notified about houses or rooms that are available for rent.
2. An intelligent application that can talk with humans and respond to their questions.
3. Using a computer to print a physical item such as a cup or a model of a building.

## Concept of emerging technologies and trends

This section describes emerging technologies that drive Fourth Industrial Revolution (4IR) and their trends. It also presents examples of emerging technologies. The section finally highlights the significance of these technologies for present and future socio-economic development.

### Meaning of emerging technologies and trends

Emerging technology is defined as new technologies that advance the interaction between computers and humans.

Emerging technology results from advancing collective knowledge and application related to methods, techniques, skills, and procedures that address current and future industry challenges and opportunities. Emerging technology has applications in many industries such as agriculture, health, education, transport, tourism, entertainment, communication, media, and energy. In most cases, technology continues to evolve due to further improvements necessitated by inventions and developments. The world is changing quickly, resulting in emerging modern data inputs, processing, storage, and output technologies. In the IT industry, technological advancements

are happening at light's speed, from stand-alone computer applications to web and mobile applications. Also, it advances from centralised applications to distributed systems to cloud-based applications. In terms of emerging technologies, the trend is changing from various spheres, including artificial intelligence and machine learning, to the Blockchain, autonomous vehicles, virtual reality, augmented reality, unmanned aerial vehicles, big data, and the Internet of things. Technological trends can also be viewed in generations such as the 1<sup>st</sup>, 2<sup>nd</sup>, 3<sup>rd</sup>, 4<sup>th</sup>, and 5<sup>th</sup>.

Typically, emerging technologies have a trend. The trend is a general shift or change in the direction of something new or different. In the context of digital advancements, a trend refers to a general direction of digital technologies, including an impact they currently make or expect to make in the future.

### Examples of emerging digital technologies that drive 4IR

Emerging digital technologies are cropping up in various industries and individual life. These technologies are artificial intelligence, machine learning, Blockchain, autonomous cars, and virtual and augmented reality. The following outlines give a quick overview of some of the emerging digital technologies:

*Artificial Intelligence (AI):* Enables computers to mimic the thinking and actions of human beings. For example, map-based location services can guide you along a route to your destination.

*Machine learning:* Computers learn from existing data and can perform new tasks based on experience gained already. For example, machine learning

allows police, to discover criminals and determine their true identities by scanning fingerprints. It is also used by immigration officials to track people crossing the border.

*Blockchain:* Emphasises building distributed and decentralised databases whose records are hard to change. For example, achieving unchallengeable records in a land registry guarantees the impossibility of illegally transferring a title deed.

*Autonomous vehicles:* Autonomous vehicles are self-driving vehicles operating independently without a human driver. For instance, the use of autonomous vehicles which promises to reduce accidents.

*Virtual Reality (VR):* This is an entirely immersive experience created in a computer-simulated environment to replace a real-life environment. It is applied in entertainment to provide near feelings simulated by computers.

*Augmented Reality (AR):* AR is a technologically enhanced representation of the actual world created by supplying additional digital information.

*An Unmanned Aerial Vehicle (UAV):* This is an aircraft that does not have a human pilot or passengers on board and is operated entirely or partially remotely by a human. For example, some small drones are used to take aerial pictures in a city or crowded areas.

Emerging digital technologies play a crucial role in the modern world by making it a better place to live. Their significance is invaluable to count, but fewer of them may be highlighted. Emerging technologies help us to do the following:

*Automate jobs that humans previously performed:* Automated devices nowadays, perform various jobs both with high risk and with a risk free to humans. Robots, for example, carry out various repetitive tasks in advanced manufacturing industries, such as painting a car or welding metals. Figure 7.1 shows a robot welding metal in a factory. They relieve humans from dangerous environments, for example, excessively hot temperatures, dangerous rays, and toxic environments.

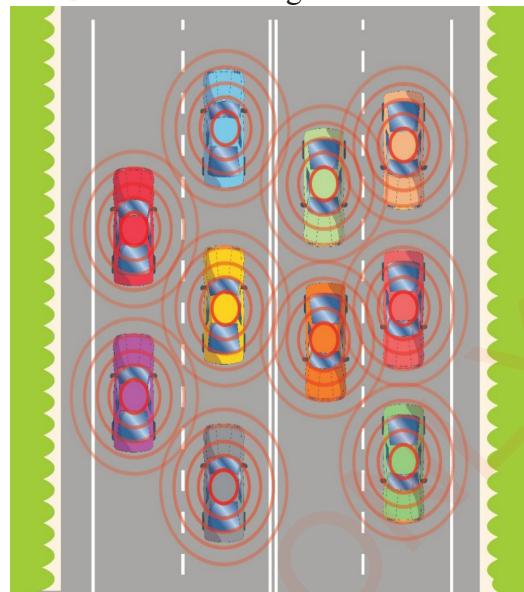


**Figure 7.1:** A robot doing its task of welding metals (Source: <http://www.automate.org>)

*Increase efficiency in productivity:* Once programmed, machines may accomplish tasks faster and more accurately than humans. For example, extensive data taken from social media messages (posts) may be analysed in less than an hour using machine learning algorithms to identify subjects and the main issues mentioned. This task might take many days for humans to complete.

*Improve safety in vehicles:* Vehicle-to-vehicle communication, shown in Figure 7.2, is expected to improve safety by allowing almost colliding automobiles to detect each other and take prompt

action before colliding.



**Figure 7.2:** Vehicle-to-vehicle communication

*Diagnose and monitor patients' health:* Wearable devices, as shown in Figure 7.3, communicate sensed health information to cloud servers to help medical doctors predict a disease state and appropriate treatment.



**Figure 7.3:** Wearable respiratory devices.

Features of emerging technologies

The following are some of the features of emerging technologies:

1. High processing power compared to traditional technologies
2. Capability to store and analyse a large volume of data within a short time
3. Advanced analytics model to generate useful information

4. Radical novelty leads to disruptive transformation
5. Automated decision-making

### Exercise 7.1

Answer the following questions:

1. Matinde, a form five student, doubts whether the Short Message Service (SMS) is an emerging digital technology or not. How can you help Matinde to clear these doubts? Explain your answer by exploring various sources.
2. Explain how Blockchain technology can be used in agriculture.
3. How do emerging technologies that drive 4IR differ from other technologies?

## Industrial revolutions

The modern world has changed an agrarian and handicraft economy to one dominated by industry and machine manufacturing. These technological changes introduced novel ways of working and living and transformed societies on how they interact.

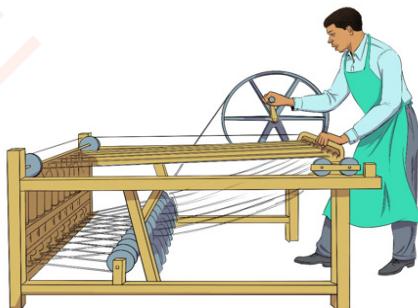
The world has experienced four industrial revolutions. These revolutions are the first industrial revolution, second industrial revolution, third industrial revolution and fourth industrial revolution.

### First industrial revolution

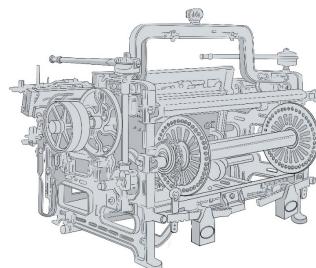
Before the first industrial revolution, hardworking people used their hands to power machines (muscle power). Production took place in people's homes. From the end of the 18<sup>th</sup> century, a new

trend emerged: hardworking individuals were replaced by steam engines that use heat as a crucial technology. Remember, steam is created when water is heated. Following the invention of the steam engine, the *first industrial revolution* began with this shift, whereby, steam was used to power everything from agriculture to the textile industry.

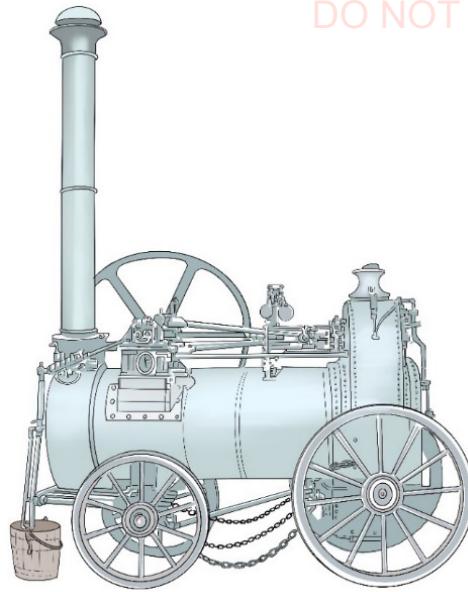
This revolution was characterised by mechanisation, where mechanical machines replaced handcraft activities in production. Examples of machines invented in the first industrial revolution powered by steam include the mechanised spinning jenny shown in Figure 7.4, the power loom in Figure 7.5, the steam locomotive shown in Figure 7.6, and the steamship in the transport industry. Also, steam power was used to pump water out of coal mines, and that steam pump was known as the “atmospheric engine”. It was generally the age of mechanical production



**Figure 7.4: Spinning jenny machine**



**Figure 7.5: Power loom machine**



**Figure 7.6:** A steam-powered locomotive



### Activity 7.1: Watching animation or videos of steam powered engines.

#### Resources:

Computer with internet connection, video playing software, pen and piece of paper.

#### Procedures:

1. Open the YouTube channel (<https://www.youtube.com/>)
2. Use the following phrase to search the animation or video: “steam powered engine”
3. Play and watch the video or animation.
4. After watching the video or animation, answer the following questions:
  - (a) What was the source of energy used to convert water into steam?
  - (b) How was mechanical energy produced?

## Second industrial revolution

The second industrial revolution was sparked by discovering new energy sources in about 1870, mainly electricity and oil. Internal combustion engines made this discovery possible, which eventually succeeded steam engines.

During this time, telephones and vehicles were also invented. Steel supplanted iron in construction because it was more durable and less costly, allowing train lines to be built for less money. With the invention of the telegraph and the telephone, communication techniques changed considerably, and transportation systems evolved dramatically from using steam to petroleum oil as crucial fuel.

Before electricity, people used wood, candles and gas lamps for lighthouses and factories. Therefore, the invention of electricity brought a new era of high-voltage alternating current. Besides the telephone invention, electricity led to other inventions such as light bulbs. Early automobiles built on combustion engines started to be sold to the public in this period. Also, this period witnessed the creation of the first aeroplane flights, as shown in Figure 7.7. Furthermore, the typewriter that computers have now replaced constituted the inventions of the second industrial revolution.

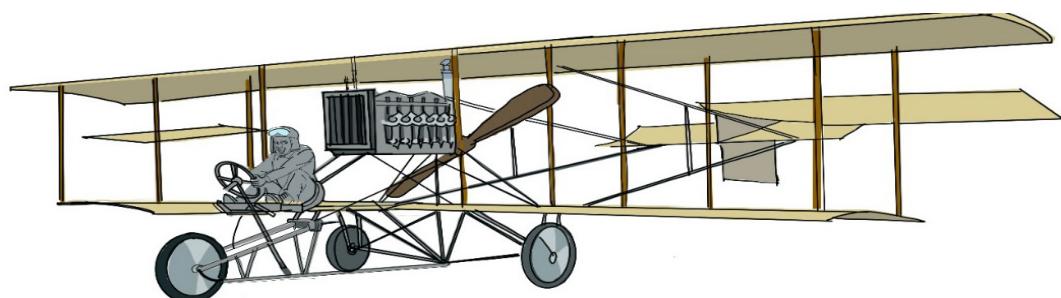


Figure 7.7: An aeroplane in the second industrial revolution

### Third industrial revolution

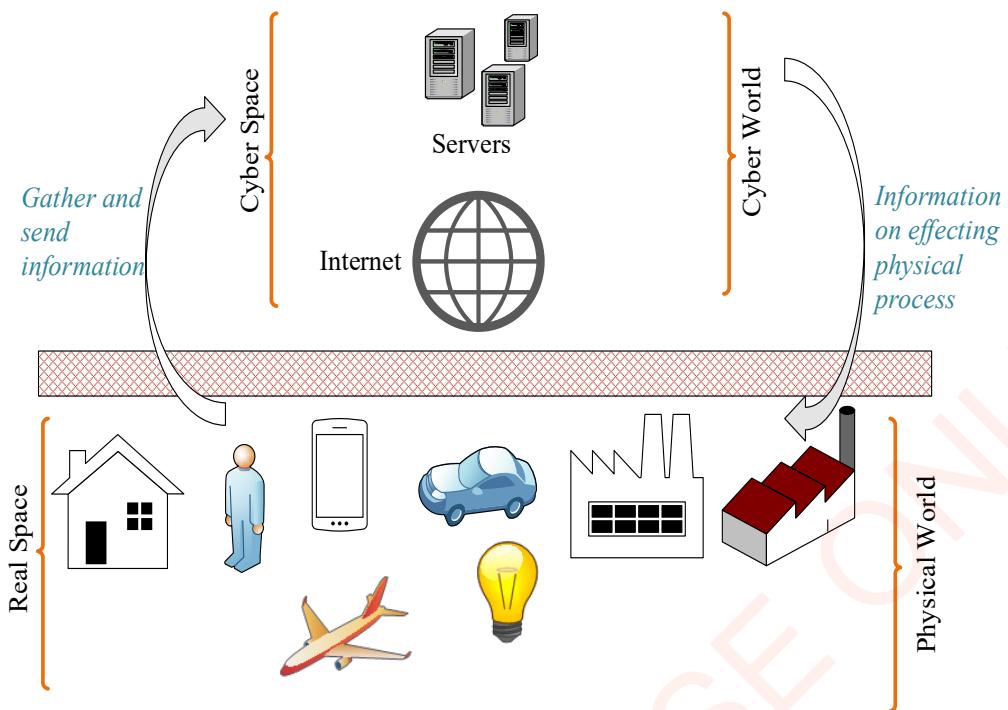
In the 1970s, the third industrial revolution began. The widespread use of electronics and computers and the introduction of the Internet characterised this revolution. Such a revolution is also known as the *digital revolution*. Semiconductors, mainframe computers, personal computers, mobile phones and the Internet were all part of the digital revolution. It was a period when world-wide-web and the use of email featured. Some manufacturing industries started to use computers to manage and control production, as shown in Figure 7.8. Many administrative tasks were now handled using electronic management information systems for organisations. These systems are still used today, although they have been improved.



Figure 7.8: Digital manufacturing during the third industrial revolution

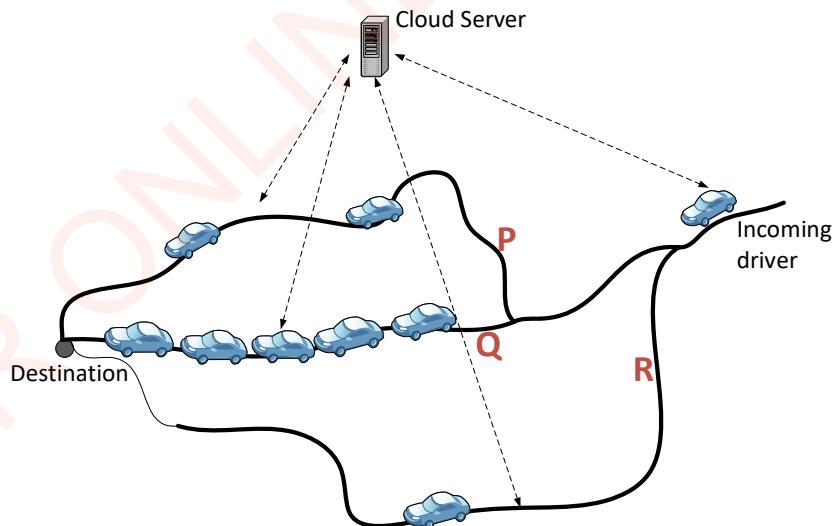
### Fourth industrial revolution

The fourth industrial revolution (4IR) appears to have started in 2010. The 4IR is characterised by Cyber-Physical Systems (CPS). The CPS are systems in which computer components are tightly linked to physical world processes while offering and using internet-based data services. Figure 7.9 shows an overview of CPS's basic architecture in the 4IR. It is composed of real and cyber-space, characterised by increased connectedness and a high degree of coupling between the two spaces. The high connectedness and coupling of real space (physical) and cyber-space (computation) in CPS are facilitated by the sophisticated process of gathering, sending and actuation of information between the two spaces.



**Figure 7.9:** An overview of CPS basic architecture in 4IR

The CPS application may be seen in Figure 7.10, which depicts a road journey. The incoming driver has three options for getting to the destination: road “P,” “Q,” or “R.” Every motorist on the road has a smartphone that is linked to a cloud server that provides access to map-based location services.



**Figure 7.10:** An application of cyber-physical systems in transport logistics

In Figure 7.10, the first part, which is a physical process, is the movement of vehicles. The second part is the cyber system, which consists of internet and cloud services containing algorithms that advise incoming drivers on which path to take. The optimal route is recommended based on the distance to the destination and the amount of traffic expected ahead. Road “R” is longer than “P,” and road “P” is longer than “Q” in this case. Algorithms on the cloud server for an application, such as Google Maps, decide which road is congested by analysing vehicle movement speeds

based on GPS data gathered from drivers’ smartphones. Slow or no speed indicates that there is a lot of traffic. Based on the remaining options and distance cost, the cloud service algorithms will then suggest to the drivers which road to take to reach their destination. In this situation, route “P” would be selected.

To summarise, Figure 7.11 depicts the development of industries from the late eighteenth century to the present day. It lays forth the path between one revolution and the next using textural descriptions and images.

|                                                                                                                                         |                                                                                                                                                                           |                                                                                                                                                                                                 |                                                                                                                                                                   |
|-----------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                                                                                                                                         |                                                                                                                                                                           |                                                                                                                                                                                                 | <ul style="list-style-type: none"> <li>• Cyber-Physical Systems (CPS)</li> <li>• IoT connected devices, network, process automation and data analytics</li> </ul> |
| <ul style="list-style-type: none"> <li>• Engines powered by water and steam</li> <li>• Mechanisation of handcraft activities</li> </ul> | <ul style="list-style-type: none"> <li>• Mass production based on engines powered by electricity</li> <li>• Invention of assembly lines powered by electricity</li> </ul> | <ul style="list-style-type: none"> <li>• Computerization</li> <li>• Use of embedded Programmable Logic Controllers</li> <li>• Automation of production processes</li> <li>• Internet</li> </ul> |                                                                                                                                                                   |
| 1 <sup>st</sup> Industrial Revolution<br><b>End of 18<sup>th</sup> century</b>                                                          | 2 <sup>nd</sup> Industrial Revolution<br><b>1870</b>                                                                                                                      | 3 <sup>rd</sup> Industrial Revolution<br><b>1970</b>                                                                                                                                            | 4 <sup>th</sup> Industrial Revolution<br><b>2010 Onwards</b>                                                                                                      |

**Figure 7.11:** The summary of key features of the Industrial revolutions



### Activity 7.2: Creating a cyber-physical system use case in fourth industrial revolution

Ms. Halima is a pastoralist who lives in Tanga but keeps livestock in Kilosa Morogoro. She employed staff to look after her cattle. Her issue is determining when cows are going for grazing and returning back home so that she can pay her staff based on the worked hours. Ms. Halima approaches you for assistance to develop a solution to that dilemma. Meanwhile, your relative who works for a large IT firm previously informed

you that there are sensors that can be connected to a physical thing and record the motion and location of the object to which they are attached. These sensors have the capacity to communicate the information they collect. Based on this scenario:

1. Identify infrastructure needed to solve this problem;
2. Use a diagram to create a design of your solution to solve this problem;
3. Devise, using a diagram, how Ms. Halima will get information related to what is going on in her farm in Kilosa in real time.

**Exercise 7.2:**

Answer the following questions:

- How did the availability of fossil fuels like coal enable industrialisation?
- Identify the impact each industrial revolution had on society.
- What is the main difference between the first and second industrial revolutions?
- Compare the third and fourth industrial revolutions and outline features that distinguish them.

the simulation of human intelligence in computers to perform tasks similar to those carried out by humans. Under AI, machines are programmed to mimic the thoughts and actions of humans, which means that those machines are enabled to reason and respond by taking actions as a response. AI systems must be able to collect information (*sensing/perceiving*), manipulate facts from newly sensed information and previous experience (*reasoning*) and make actions (*responses*) mainly on their own.

On the other hand, machine learning is a branch of computer science that focuses on using algorithms and existing data to build a model that can predict an outcome of something. In creating the model, the algorithm has to learn from historical data. In that learning, the algorithm is calibrated several times until its accuracy or prediction level is confidently sufficient. Once the algorithm is ready and accurate, it may be subjected to a related problem with a new data set to provide the desired prediction.

### **Importance of artificial intelligence and machine learning**

AI systems have a critical relevance in our daily lives in various ways. First, an intelligent machine can function in a risky environment without harming itself; thus, relieving many of the constraints placed on humans. Second, humans are prone to errors, mainly while doing repetitive tasks. Unlike humans, intelligent machines are unique in that they seldom make errors.

AI systems can make better judgments faster because they can examine and analyse all available options before

## **Emerging technologies and their applications**

In this section, you will learn about emerging digital technologies that drive 4IR and their importance. Next, you will learn about the application of each technology in daily life. Digital technologies described in this section are Artificial Intelligence, Blockchain, cloud computing, 3D/4D/5D printing, Big data, and the Internet of things. Other technologies are drones, virtual and augmented reality, autonomous vehicles, robotics, and 5G.

### **Artificial intelligence and machine learning**

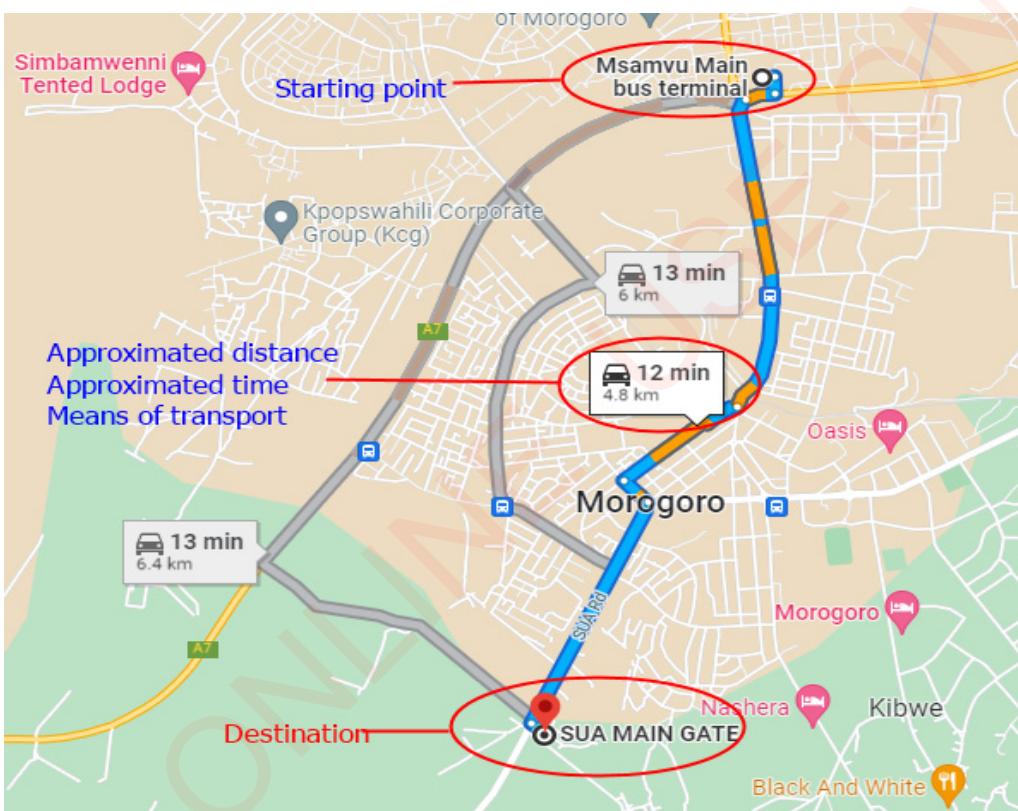
John McCarthy first coined the term Artificial Intelligence (AI) in 1956 at the Dartmouth conference. John McCarthy defined Artificial Intelligence (AI) as the science and engineering of making intelligent machines. AI is a technique for getting machines to work and behave like humans. AI applies contrived produced intelligence to solve problems in the real world. AI comprises

arriving at a final decision. The situation is different when humans are involved in deciding with many choices. When given more than seven alternatives to choose from, most people find it challenging to go through them all and make an informed decision. Additionally, AI systems do not require vacations or pay, and they do not become ill or get tired.

### Application of artificial intelligence and machine learning

AI has applications in many areas, including:

- Health:* To support diagnose illnesses and provide professional guidance (expert systems)
- Travel and transportation:* Supplying pedestrians, car drivers and airlines with map-based location and navigation services. Figure 7.12 shows AI-based google location service in a selected route.



**Figure 7.12:** A Google location service in a selected route

- Business:* For offering online client conversation, for example, through Chatbots.
- Facial recognition:* Facial recognition is used to recognise people's faces at places like immigration offices and airports and for individual use like logging into laptops and accessing other facilities.
- Audiobooks:* To listen to a book read aloud by the author. Instead of reading text from a book (visual), the intelligent system will read the book to you in audio

format, allowing you to hear the voice. You can read a book while riding a bicycle, just as you can listen to music while driving a car. It can also benefit visually impaired people to access such material services for learning, working, and any other purpose in their lives.

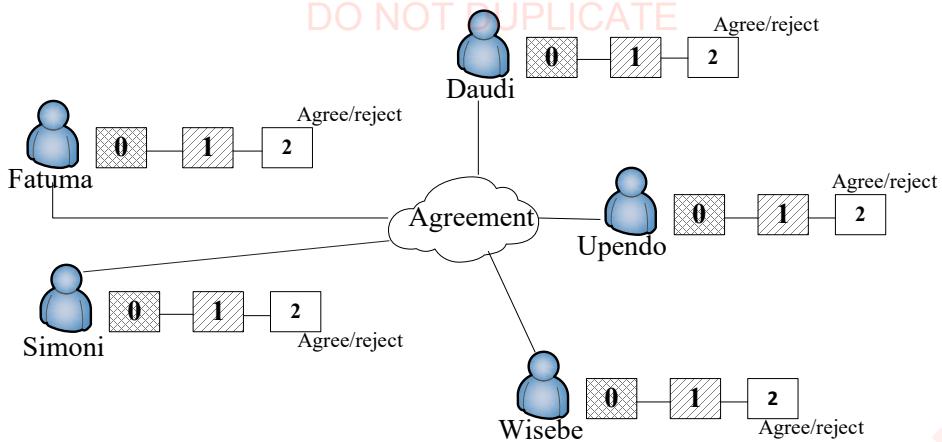
### Blockchain

Blockchain technology is a distributed electronic ledger of digital records, events or transactions that are cryptographically secure, extremely hard to forge, and updatable through a consensus protocol agreeable to all connected nodes. A Blockchain is a decentralised, secure digital record of time-stamped, unchangeable transactions in several copies on internet-connected computers. That copy is constantly accessible to everybody at any time, and users modify it via distributed agreement. Blockchain technology focuses on creating distributed and decentralised databases with immutable records. It does not allow for the deletion or updating of data; instead, new data is added to the preceding to create a block of data connected in a chain. Instead of depending on a central authority to handle everything, Blockchain technology allows everyone to keep records maintained in a ledger. As a result, each network member has equal

control over records. It's used to record and keep track of transactions and manage assets in a corporate network, including real estate, money, and machinery equipment.

Consider Village Community Banks (VICOBAs), where community members join together to save money and obtain credit. There are already mobile platforms where members may submit their monthly savings using mobile phones. Financial savings are centralised and administered by a governing body, usually a member selected from a group. Other members may access the information, but only with restricted access. A client-server network with a central database characterises this type of record-keeping.

In a Blockchain, however, each network member has an equal ability to govern records. That Blockchain consists of a distributed peer-to-peer network in which each member maintains their records (ledger). Figure 7.13 shows the network members of the VICOBAs who are Fatuma, Simoni, Daudi, Upendo, and Wisebe. All these members must agree to record a transaction in block zero (0). If they agree, that successful transaction will be time-stamped. If a transaction has to be updated, a new block of data, say block one (1), is created and connected to block zero instead of altering it.



**Figure 7.13:** An example of a distributed peer-to-peer network for VICOBIA savings

All members of the peer-to-peer network must agree to commit to the transaction to be successful. Each member is in charge of their record (ledger). No data block may be erased, resulting in immutable records. In Figure 7.13, subsequent data blocks are linked, producing a chain of data blocks, hence the term Blockchain.

### Importance of Blockchain

Blockchain is useful because it helps guarantee trust in business networks, which traditional business-to-business systems cannot provide. For example, with a shared database, there is no guarantee that an administrator has not transpired with the transaction. Blockchain gives the participants of the transaction non-repudiation, which is a piece of evidence that the transaction was agreed upon. Other importance of Blockchain are as follows:

**Improved security:** Blockchain helps prevent fraud and unlawful behaviour by producing a record that cannot be changed and is encrypted end-to-end. Any effort to change a transaction will

fail because other nodes will deny a request to publish transactions to the network only because each node has a copy of all transactions completed.

**Increased transparency:** Transactions and data are recorded instantly and identically in many locations because of distributed ledgers, allowing all network members with permission access to see the same information simultaneously.

**Reliability:** The possibility of failure is scarce because multiple copies exist across the network.

### Challenges of Blockchain

Some challenges with Blockchain technology need attention before being used for transactions. The following are some challenges of Blockchain:

**Privacy:** There is a possibility that someone could track down a person's identity in the network through transactional data, just as web trackers and cookies are generally used by businesses. This proves that Blockchain is not 100 percent secure.

**Scalability:** The practical use of Blockchain becomes a bit hard to imagine with the issue of scalability in view. Each participant node must verify and approve a transaction, so one Bitcoin exchange can take up to several hours.

**Security:** The attack can be implemented if the nodes in a network lie; the lie will be accepted as truth. Therefore, everyone in the network will have to continually have a lookout for it to perceive any unwanted influence.

**Storage:** Since Blockchain databases are stored indefinitely on all network nodes, the issue of storage surfaces. With the increasing number of transactions, the size of the database will only expand, and there is no way personal computers can store unlimited data that just gets appended.

**Regulations:** Regulatory organisations in the financial field are a challenge for Blockchain implementation. Blockchain applications will have to lay down the process of locating the offender in case fraud occurs, which is a bit challenging. Blockchain technology's other regulatory aspects will need to be introduced first to facilitate its broad implementation.

### Application of Blockchain

Blockchain has many applications, including:

1. *Cryptocurrency:* People and organisations can send and receive money without central intermediation or banks. Examples include Bitcoin and Ethereum.
2. *Central Bank Digital Currency (CBDC):* The CBDC is a digital

form of central bank money widely used by the public. Unlike cryptocurrencies, they are based on a country's agreement currency value and are developed in many nations. The CBDC promotes safety, robustness and efficiency of payment, reduces cost, and increases transaction convenience.

3. *Smart contracts:* Smart contracts refer to the computer code shared between participants of the business network, and these implement the business rules associated with each transaction. As the code is shared, it can be executed by all relevant, and they can agree on the output. Contracts are maintained digitally on the Blockchain and are automatically executed when particular criteria are met.
4. *Property registry:* In a property registry, unchangeable records are kept to determine how future ownership transfers were completed from the initial owner to subsequent owners.

### The characteristics of Blockchain

**Secure:** each transaction recorded in the database is digitally signed and mathematically guaranteed to be authentic and impervious to fraud.

**Shared:** The database's primary value is shared between separate entities. The more entities shared, the greater their value.

**Distributed:** Many copies of the database exist, replicating each other in relative time.

**Ledger:** A write-once, thus immutable ledger that records every transaction. The entire history of the whole transitions is available mathematics does not lie, and there is nowhere to hide.

### To illustrate how shared ledgers and smart contracts work, imagine the scenario of a Blockchain that tracks a car's ownership

The Blockchain itself would be an ordered data structure that contains the details of every ownership change agreed upon. For example, one transaction could state, "the car's ownership with identification number T576 CKY has changed from Pendo to Asha. The smart contract associated with this transaction describes the computer logic that makes the transaction happen. For example, check that the seller is equal to the current owner, and if the owner is set to be the buyer, decrement the buyer's cash balance and increment the seller's cash balance. Each interested participant, Pendo and Asha, would run that smart contract code and agree on the output, and if everything is acceptable, then updates the ledger accordingly. The updated ledger might be made visible to Pendo and Asha but all interested parties, such as vehicle regulators or insurers, pending the agreed rules of the network.

### Cloud computing

We live in a digital age where one of the most critical issues is accessing service from seamless nodes. Have you ever anticipated keeping files in a remote storage facility such that you can access

it anywhere on demand? Your data is held in the cloud, being a storage service provided by cloud computing. Cloud computing is the term that describes any service delivered over the Internet. A motivation behind cloud computing is to provide computing services outside a locally installed computer or computer network, generally outside a local premise. These computing services, typically hardware and software, are usually delivered through the Internet by third-party companies. Services like remote storage of files, online application software, and virtual websites and databases are all happening under the facilitation of cloud computing technology. However, the challenge to remote storage is a lack of security, since the stored data is in the third part and cannot be controlled.

There are three types of services offered in cloud computing; categorised as, infrastructure as service, software as service and platform as service.

*Infrastructure as a Service (IaaS):* This sort of cloud computing is concerned with providing hardware services such as real and virtual servers on demand for various applications such as, hosting and storage. IaaS is commonly used to host a website or a mobile application. In the case of non-free IaaS, the membership price is usually paid monthly or according to any other arrangement negotiated.

*Software as a Service (SaaS):* This type of cloud computing is concerned with providing software as a service. Instead of installing software on your computer, you access it over the Internet and utilise it in your daily activities on demand.

SaaS includes web-based email (such as Yahoo and Gmail), collaboration software such as Google documents (Google docs) and file storage such as Dropbox and Google drive.

*Platform as a Service (PaaS):* This type of cloud computing service provides tools for developing and deploying applications that run on a remote server. An example of PaaS is writing, compiling, and running computer programs using online compilers (such as C++ compiler).

### Importance of cloud computing

Cloud computing services are helpful in a variety of circumstances. Some of the benefits are:

*Backup and restoration:* It is easy to back up your data offsite and restore it if something goes wrong.

*Low cost:* It allows you to save a lot of money because it doesn't require any hardware.

*Security:* Cloud security covers the set of controls, policies, technologies, procedures, and services that ensure the safety of the hosted services over the Internet and protect cloud data and infrastructure. The benefits of the rapid operation, flexibility, and scalability have made cloud computing famous among many business organisations. Cloud computing offers state-of-the-art security mechanisms that guarantee data is stored and handled safely.

*Easiness in deploying an application:* It is easy to deploy an application you developed because you do not need to buy your server, manage and maintain it.

*Convenience:* Access to cloud-based services may be done anywhere, provided you have access to the Internet.

### Disadvantages of cloud computing

Despite the benefits of cloud computing usage, it is undeniable that this system also has some disadvantages such as:

*Internet connection dependent:* The Internet is the only way to cloud computing. When there is no Internet connection in your place or the Internet path to the cloud, the provider is in trouble, then, the cloud computing machine is disconnected. The big challenge in developing countries and remote areas is that they do not have reliable internet access. The weakness of the public cloud is that everyone accesses the same server, which will increase the risk of attack and put down the server.

*Data confidentiality:* There is always a risk that other people can access user data. So, data and cloud protection must be good because it will not be dangerous for data confidentiality.

*The level of security:* Secrecy and security are among the most suspicious things in cloud computing. Using a cloud computing system means we are entirely entrusted with data security and confidentiality to companies that provide cloud computing servers. When you experience a problem, you cannot sue the server for errors in the data.

*Data mobility:* It refers to the possibility of sharing data between cloud services and how to retrieve data if one day the user terminates cloud-computing services, and there is local storage where the data can be used at any time as needed. If cloud-computing services is unavailable, the users will suffer if no other local storage is available.

*Vulnerable to attacks:* Each component of cloud computing can be exposed to a wide range of attacks. This is a wide-open opportunity for attacks on data or activities stored on the cloud server.

*Technical problem:* Cloud computing makes you unable to manage technical issues on yourself. When you face a problem, you must contact customer support, who is not always ready. This is a problem because some cloud support services are not free.

### Application of cloud computing

There are many applications of cloud computing services in different sectors. These applications are such as:

1. Online learning in education, where information portals and platforms are easily deployed. Then, students can attend virtual classes, online examinations, and even scholarly discussions. Moodle platform is one of the examples of an e-learning platform that is commonly operated on the cloud.
2. Online tools are available across various fields, like in graphics, tools such as, image editing, for example, by removing the image's background. Examples of online image editing tools include photopea, adobe online photo editor, pixlr X, pixlr E, etc.
3. Converting documents format: You can convert a document from one format to another, for example, from Pdf to word. Examples of this software are such as PDF converter, and Nitro.

4. **Social media:** You keep connected with your friends through social media. For example, Facebook, WhatsApp and Telegram.

5. **Electronic meetings:** It is common now to see people holding video conferencing meetings. You communicate with a friend through video calls, even at an individual level. An example of an electronic forum is Zoom in teaching and business meetings. Also, people use WhatsApp video calls and Zoom to talk to each other on individual levels.



### Activity 7.4: Converting format of a document online

#### Resources:

A computer connected to the Internet, note book and pen

#### Procedures:

1. Open a search engine such as Google (<https://www.google.com/>).
2. Use the following phrase to search: "online conversion of a file".
3. Open a link to an online file converter that you prefer.
4. Follow instructions provided in that specific online file converter to change a file format, for example from Pdf to word.
5. List down any four document converters from your search.
6. Try one after another converter to convert a document to a format of your choice.

7. After changing the file format, answer the following questions:
- Assess the performance and quality of each document converter.
  - Did a layout of your document remain intact or was slightly changed?
  - What are your opinions on this online service?
  - Is this service a SaaS, IaaS, or PaaS?



### Activity 7.5: Removing background of an image

#### Resources:

A computer connected to the Internet, notebook and pen

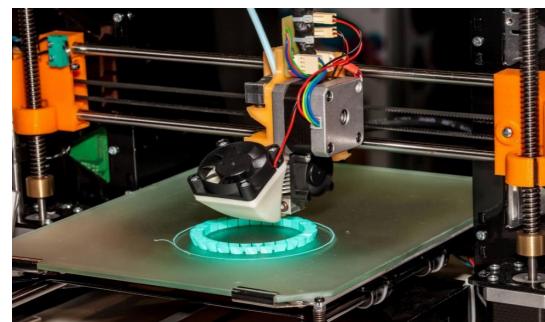
#### Procedures:

- Open a search engine such as Google (<https://www.google.com/>).
- Use the following phrase to search: “remove background from image online”.
- Open a link to an application of your choice that removes background of an image.
- Follow instructions provided in that specific application
- After removing background of the image, answer the following questions:
  - Did the quality of your image maintained or was slightly changed?
  - What are your opinions on this online service?
  - Is this service a SaaS, IaaS or PaaS?

### 3D printing

Consider a scenario where your computer connected to a printer can create tangible things such as a cup. 3D printing is a method of turning a computer-generated design into a three-dimensional object. Typically, printers that are used to print material things are not the same as the ordinary printers we know and use in our workplaces and homes. Unlike the regular printer that type-sets characters and images on paper, the 3D printer works by layering (adding) new layers on top of the old ones. Figure 7.14 shows an example of a 3D printer.

Initially, an object to be printed is designed on a computer using computer-aided design (CAD) software. Next, that logical design is commanded for printing. Printing is done using materials heated to a particular temperature to be flexibly used to build an item in the form of layers. As technology continues to grow, other generations of 3D printing are emerging. Although small things can now be printed, this technology is still growing. The world is now considering advancements in 4D and 5D printing.



**Figure 7.14:** An example of a 3D printer

4D printing is an extension of 3D printing. In 4D printing, an item is given the ability to change its form or structure predictably over time after it has been

printed. Such structure changes after exposure to various influential stimuli such as temperature, light or other environmental triggers. Moreover, 5D printing uncovers the limitations of 3D printing by printing physical objects with curved surfaces instead of flat layers in 3D printing. Usually, in 3D printing, a print head moves while the printed object is stationary. Instead, with 5D printing, the print head is fixed while the printable thing is moving. Turning the printable object at different inclination angles enables it to yield curved surfaces.

### Importance of 3D printing

3D printing is relevant in many areas, including:

*Rapid prototyping:* You can design your product and print it in a shape that is easy to see what it looks like in physical shape. Then you can continue to improve the design and print again until a viable product design is obtained.

*Creation of personalised items:* Due to consumer preferences, the same product may have minor differences, which is a one-of-a-kind item. A simple example is a cup with your name and photo on it. Producing 100 customised items will simply require you to know the preferences of each buyer, and printing will be simple.

*Enables waste elimination:* Waste elimination becomes possible because an item is printed according to a shape designed, thus no need to cut out excess parts and throw them away.

### Application of 3D printing

The 3D printing has many applications in industry, including:

- (a) 3D printing can be used to produce water pipes in a plumbing system. Future water pipes will be able to modify their structure in response to changes in water flow or temperature, preventing them from bursting.
- (b) In the medical field, 3D is used to create prosthetics and tooth implants and may be used to print human organs in the future.



### Activity 7.6: Watching videos about 3D printing

#### Resources:

A computer connected to the Internet, pen, and notebook

#### Procedures:

1. Open the YouTube channel (<https://www.youtube.com/>)
2. Use the following phrase to search video: “3D printing”
3. Watch more than three (3) videos
4. After watching the videos, answer the following questions:
  - (a) Explain how layers are laid down to one another
  - (b) In each video, what product was printed out?
  - (c) Demonstrate by describing procedures on how 3D printing work
  - (d) Evaluate the benefits of 3D printing compared to traditional approach?

In the future, 3D could print local and on-demand medicines according to specific patients' needs.



### Activity 7.7: Watching videos about 4D printing

#### Resources:

A computer connected to the internet, pen, and a notebook

#### Procedures:

1. Open the YouTube channel (<https://www.youtube.com/>).
2. Use the following phrase to search video: “4D printing”.
3. Watch more than three (3) videos.
4. After watching the videos, answer the following questions:
  - (a) What is the difference between 3D printing and 4D printing?
  - (b) In each video, what stimuli caused a material to change its structure?
  - (c) Reflecting back to our individual lives and in industries in Tanzania, identify at least four (4) things we can employ structures that change shape after being exposed to external influences.

### Internet of Things (IoT)

The Internet evolved from a network of computers to a network of computers and people. We are now entering the world of the Internet of things beyond computing and network devices, including animals, household appliances, and cars, to name a few. As a result, such typical Internet has earned the name “*Internet of Things*.” The Internet of Things (IoT) is a network of linked devices that includes machines, goods, animals, and people that can exchange data without human interaction. Instead of having the Internet only for people and computers, other things like smart refrigerators, television, car and microwave are equipped with small devices that can sense information and communicate it to other devices or computers, hence, the Internet of things and the Internet of everything. Figure 7.15 shows an illustration of the Internet of Things.



**Figure 7.15:** An illustration of the Internet of Things

**Imagine the following scenario**

*A person going to work while wearing an automatic blood pressure machine that measures blood pressure periodically and send results to a medical doctor. Being late at work, that person uses a smartphone to switch on light at home. After a while, the person receives alerts from motion sensors installed in the premise at home. These warnings indicate that a thief or unauthorised persons are near the house's fence. Having forgotten to water flowers at home, moisture sensor installed in a garden sends alert to mobile phone, informing that flowers need watering. That person uses the smartphone to open water tapes for a given period. Planning to get back home, and it is a day with hot weather, that person uses smartphone to switch on air condition at home to cool temperature, so that on arrival, the house will already be cool. Before arriving home, a refrigerator sends alert to the smartphone that soft drinks like juice are out of stock. That person opens website of a store or supermarket and order for juice.*

This scenario elaborates on a storytelling about how IoT changes how people control their surroundings. Things like soil, refrigerators, houses, light bulbs, and water tapes are embedded or attached to devices that send and receive information to/from cloud computing services. In this context, the Internet is no longer limited to people but extends to many things, including livestock farms and appliances.

### **Importance of the Internet of Things (IoT)**

The Internet of Things enables information gathering, transmission, and storage for devices in various scenarios, creating or accelerating the

development of many applications. IoT includes industrial control systems, the retailing industry, innovative shelf operations, healthcare, food or restaurant industry, the logistics industry, the tourism industry, and academic libraries applications.

Nonetheless, it is expected that the Internet of Things will significantly contribute to addressing crucial issues such as marketing strategies, public health care monitoring systems, everyday living monitoring, and traffic congestion control.

The increased usage of IoT devices is bringing importance to a variety of fields, including:

- Providing reader statistics in a particular industry that may be analysed to acquire more profound insights into a topic of interest;
- The possibility for remote control and monitoring of systems is possible. You might be able to keep up with what's going on at home, office or farm even if you're not around;
- Monitoring of machine performance: Sensors attached to or installed in a machine report on its efficiency.

### **Application of the Internet of Things (IoT)**

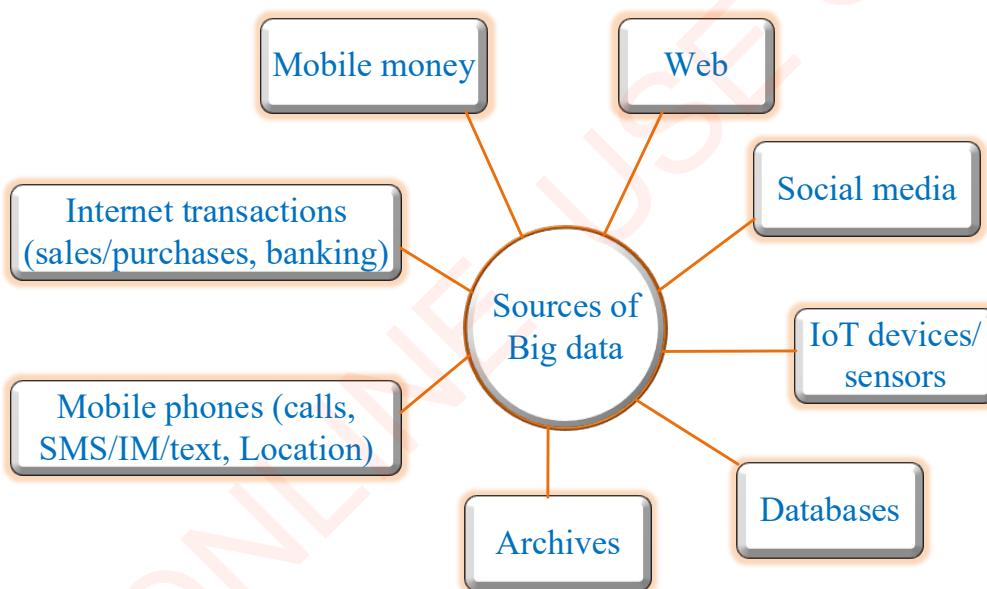
IoT can be applied in almost all areas in life. Some of the areas where IoT is widely applied are:

- Home appliances, for example, cooking appliances, refrigerators and lighting
- The smart security systems that detect unusual movement and entry into a premise
- The wearable health monitoring equipment that collects and communicates data about our health to a remote cloud server for

- processing and further analysis
- (d) Smart containers for sensing the state of commodities parked in shipping containers
  - (e) Transport logistics for tracking the movement of cargo in shipping
  - (f) Traffic monitoring whereby mobile phones report our pace of movement on the road. A slow pace from many devices implies the presence of traffic congestion
  - (g) Irrigation for sensing and controlling necessary watering required in a farm

### Big data

Data is created in large volumes and at a high rate in today's environment. For example, mobile phones that use map location services generate a daily history of our movements. The amount of data generated by global social media posts is enormous. Yet, vehicles and aeroplanes create and share data while on the route in transportation. Social media, sensors incorporated in machinery equipment (such as machines in the manufacturing industry), transaction processing systems (such as mobile money transactions), and IoT devices generate a large quantity of data, as shown in Figure 7.16.



**Figure 7.16:** Some sources of Big data

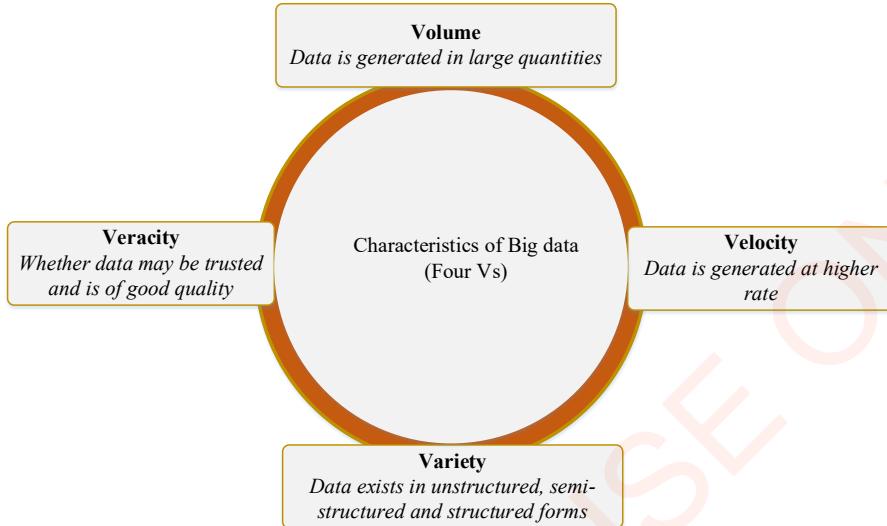
In line with the characterisation presented in the previous paragraph, Big data refers to data that is often large in volume and generated at a high rate. Big data has four primary characteristics: *volume*, *velocity*, *variety*, and *veracity*. These characteristics are also referred to as the four Vs of Big data. The following is a brief description of Big data characteristics as shown in Figure 7.17:

**Volume:** A large volume of data is generated every minute. Since the growth of information is so rapid, then new types of database management are needed.

**Velocity:** This is the immense speed at which data is created and processed. Big data is usually generated at a higher rate.

**Variety:** Data is generated from various sources. Data generated may be unstructured, semi-structured or structured, resulting in data that exists in a variety of forms.

**Veracity:** Signifies the trustworthiness of data, whether the data is correct and of good quality. The data gathered may be of both excellent and lousy quality.



**Figure 7.17: The four characteristics of Big data**

### Importance Big data

Big data has relevance in many areas, including:

- Making informed decisions due to the availability of rich insights generated from a large volume of data;
- Acquiring citizen opinions on a particular topic for smooth administration of the government;
- Empowering a better understanding of customer needs in the retail industry;
- Determining and managing risks in credit facilities provided by financial institutions.

### Big data analytics

Now that you have that Big data, the

next step is to make sense and meaning out of it, a point where Big data analytics plays a critical role. Big data analytics is the process of examining large amounts of data with advanced techniques to find hidden patterns, correlations, market trends, customer preferences, and other valuable insights. Advanced data analysis techniques are utilised to understand better issues that would take a long time to uncover if advanced data analytic techniques were not applied. Depending on the purpose, there are four main types of data analytics, which are:

Descriptive data analytics that focuses on answering questions about what happened in the past;

- Diagnostic data analytics which is designed to explain why something happened;

- (b) Predictive data analytics that looks at what could happen in the future; and
- (c) Prescriptive data analytics assists in determining what should be done in terms of decisions and actions.

### Application of Big Data

Application of Big Data include but not limited to:

- (a) Tracking customer behaviour
- (b) Recomendeshen
- (c) Smart and secure traffic

### Unmanned aerial and smart vehicles

A drone is an aircraft that has no pilot inside it. Drones are sometimes known as *Unmanned Aerial Vehicles* (UAVs). Drones differ in size, and they range from large ones, such as military drones carrying bombs and attacking in combat, to small ones used to take video and still pictures of a scene. Drones are usually controlled using a remote, as shown in Figure 7.18. Small drones have many components, including a GPS module that offers location services, a battery to power the drone, and an antenna and receiver for communicating signals.



**Figure 7.18:** A drone for spraying rice fields to fight malaria in Zanzibar

(Source: <https://apnews.com/article/b2131aa51d4b4043a7c18e6923646447>)

An autonomous or smart vehicle is a vehicle that can drive itself from a starting point to a predetermined destination in *autopilot* mode. Autonomous cars, also known as self-driving vehicles, perform driving functions on themselves without human interaction. They can sense surroundings, accelerate speed, and apply brakes to reduce speed or stop the car.



### Activity 7.8: Watching videos about autonomous vehicles

1. Open the YouTube channel (<https://www.youtube.com/>)
2. Use the following phrases to search video: “autonomous cars or self-driving cars” and “Unmanned Aerial Vehicles”
3. Watch more than six (6) videos, three for drones and others for “soft driving cars”
4. After watching the videos, answer the following questions:
  - (a) How does drones and self-driving car see objects and obstacles on their paths?
  - (b) What challenges could drones and self-driving automobiles face as a result of infrastructure flaws?
  - (c) What are the benefits and drawbacks of using drones and self-driving cars?
  - (d) Can our road infrastructure support autonomous vehicles? Explain why.

## Importance of unmanned aerial vehicles

Drones and autonomous vehicles have relevance in today's industry and society. The following list describes some of the reasons why drones and autonomous vehicles are essential.

- (a) UAVs are easy to deploy and operate, and their cost is relatively low;
- (b) UAVs can reach more difficult heights and areas than where humans can not;
- (c) The use of autonomous cars is expected to reduce accidents, particularly accidents whose sources are driver's fatigue, laziness and distraction; and
- (d) Autonomous vehicles will reduce traffic jams because human drivers have stop-and-go behaviours while driving. Self-driving cars will not exhibit stop-go behaviours, which will lead to low traffic jams.

## Application of unmanned aerial vehicles and autonomous vehicle

Drones and autonomous vehicles have many applications, including the following:

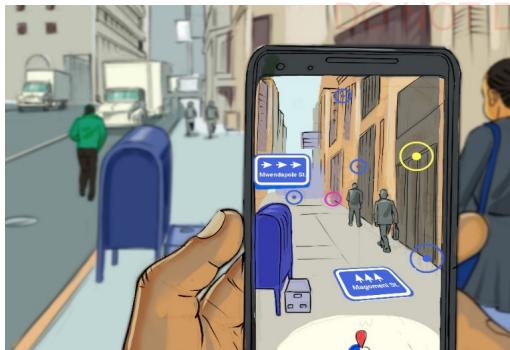
- (a) UAVs are used in geographic mapping. Drones can be applied to take images in regular and challenging areas, such as locations that are difficult to reach (mountaintops and islands)
- (b) UAVs can be used to take aerial view photos and videos in various including townships with tall buildings

- (c) Drones can be used to deliver shipping where items are moved faster and efficiently from store to a customer
- (d) Large drones can be used to perform military operations
- (e) Autonomous vehicles can be used as:
  - (i) Taxis in cities and towns;
  - (ii) Shuttles to serve need public transport; and
  - (iii) Delivery vehicles where customised delivery is even more suited.

## Virtual and Augmented reality

Virtual Reality (VR) is a term that describes a situation in *near reality*. Virtual Reality (VR) uses computer technology to create a simulated environment. That simulated environment is unreal, but it is undoubtedly near reality. VR attains this reality by enabling users to view a world through a virtual reality headset, making the user immersed in the virtual world.

**Unlike VR,** Augmented Reality (AR) is a technologically enhanced representation of the actual world created by supplying additional digital information in audio or visual components or even textural description. AR enhances your surroundings by adding digital features to a live view, most commonly via a smartphone's camera. An example of an AR application may be found in retail shopping. A customer may use a smartphone's camera to view a street, as shown in Figure 7.19. Then names of shops available in that street appear on the customer's live view, together with; a distance to a specific shop, ratings, and the number of votes.



**Figure 7.19:** Using AR to find nearby shops

### Importance of virtual and augmented reality

On the one hand, VR immerses users in a near-real world, allowing them to live and participate in a previously unimaginable setting. The VR transports the user to a new experience obtained at a lower cost. Users can study or gain a deeper understanding of a situation in prior. On the other hand, AR allows users to see actual objects broadly. It empowers users with required details when additional information is added to a physical scene or item.

### Application of virtual and augmented reality

Virtual reality and augmented reality have a wide range of applications. The following are descriptions of a few of these applications:

Application of VR are as follows:

- Virtual reality in health care: A medical doctor can practice doing the surgical operation through training without being in a real operational theatre.
- Virtual reality in entertainment: It may be used at theatres to mimic

comedy film experiences and allow people to enjoy them.

- Virtual reality in the military. Virtual reality combined with sights and sound may replicate dangerous training scenarios to prepare and train soldiers.

Application of AR are as follows:

- Augmented reality can be applied in picking products or parts stored in warehouses.
- Whether walking or driving, augmented reality provides guided navigation or guidance. It is beneficial while visiting a new location with which you are unfamiliar. The augmented reality may use the smartphone's GPS to offer you directions that you may view in real-time until you arrive at your location.



### Activity 7.9: Watching virtual reality videos

#### Resources:

A computer with video streaming software, internet connection, speaker/headphones

#### Procedures:

- Open the YouTube channel (<https://www.youtube.com/>).
- Use the following phrases to search video: "virtual reality in medical care".
- Watch more than three (3) videos plug you headphones to reduce noise.

4. After watching the videos, answer the following questions:
- Identify physical objects in the videos
  - Identify virtual objects in the videos.
  - Explain how virtual and physical objects interact.
  - Discuss how this can be applied and benefit our community.

## Robotics

Robotics is a term that describes the designing, construction, and use of machines (robot) to perform tasks that are traditionally done by humans that can interact with the world. Typically, there are two groups of robots –*physical robots* and *software robots*.

As shown in Figure 7.20, physical robots deployed in industries have five main components: sensors, control systems, actuators, power supply, and end effectors.

Using a variety of sensors, the sensory component gathers information about the surrounding environment. Data collection includes visual data (using a camera), audio (using a microphone), pressure, light, and temperature. The control system analyses data collected by sensors and decides on the best course of action. After analysis and decision, the actuator components of a robot do the robot's real job. These actuators are also known as effectors in some cases. The robot must be powered by electricity to work, which is the robot's fourth component. The end effector's last component comprises

physical components that are generally interchangeable, such as spray for the painting robot, electrode and electrode holder for the welding robot, and grasping claws for a robot that picks items.



**Figure 7.20: Robots in the car manufacturing industry**  
(Source: <https://media.jaguarlandrover.com/2016/manufacturing-success-story>)

Software robots are computer programs that perform non-physical jobs. Examples of these robots are search engines (that you use to search for various information on the Internet) and chatbots. Chatbots are known for pop-ups when you open a particular website, asking visitors about what they can do to help. They work as employees of the customer service department in an organisation.

## Importance of robots

In general, robots do tasks autonomously or semi-autonomously. Semi-autonomous robots require human intervention to complete their jobs, whereas autonomous robots do not. Some robots are intelligent, but others are not. Although it was programmed

with a pre-defined set of instructions, the robot that picks and transfers goods is not necessarily intelligent. On the other hand, an intelligent robot replicates human behaviour by perceiving information, reasoning from facts, and taking actions after a decision; take, for instance, Robot Sophia. Robot Sophia is realistic humanoid robot capable of displaying human like expressions and interacting with people.

### Application of robots

Robots may be applied in more expansive ranges areas, as follows:

- Robots are a fantastic alternative for investigating locations that people cannot easily access, such as deep-ocean research and space exploration.
- Robots are used in the manufacturing industry, for example, in car manufacturing industries.
- Robots are used in homes, for example, for doing cleanliness or cooking.
- Robots are used in the health industry, for example, in routine tasks and surgery.

### The 5G

The 5G refers to fifth-generation wireless technology in cellular telecommunication. It's important to know what came before 5G to comprehend it. The first generation –1G networks—allowed people to make phone calls, but they had a lot of flaws such as reliability and signal interference. The 2G network was based on Global System for Mobile (GSM) communication. It enabled circuit-switched mobile data

services such as Short Message Service (SMS). It incorporated packet switching in the form of General Packet Radio Service (GPRS) and Enhanced Data rates for GSM Evolution (EDGE). The 3G provided critical core network speeds for smartphones. The 3G had a four times 2G data transfer capacity, allowing video streaming and video conferences. The 4G provides high-quality data transfer speed, enabling video streaming and other related services which demand higher data transfer.

The 5G technology is the next generation of wireless telecommunications for mobile internet connections. It has a fast connection and will allow smart homes, automobiles, health care, and nearly anything to connect to the Internet, creating an integrated environment. The 5G is offers 100 times more data transfer capacity than the 4G. The facts changing the digital world is influencing the demand for high Internet speed, which may not be achieved from the traditional connection architecture of third and fourth generations. The 5G addressed three critical challenges; ultra-high throughputs, ultra-large connection, and ultra-low latency. Generally, 5G has three main advantages:

- It provides enhanced mobile broadband of up to 20Gbits/s
- It is a massive machine-type communication that has one million connections per second per square kilometre
- It has an ultra-reliable low latency of 1ms

## Importance of the 5G Network NOT D

The 5G network transforms many sectors and individuals' lives due to the network's unimaginable speed and capacity. The following are among the significance of the 5G network:

*Supporting more interconnectedness in the future:* The continued increase in the number of devices connected to, will hardly be supported with 3G and 4G in future. In contrast, 5G can support around one million devices connected to a network at ultrafast speed, within an approximate area of one (1) square kilometre.

*Speed of data transfer:* The 5G offers high speed than its predecessors. The 5G is faster than 4G, sending up to 20 Gigabits-per-second (Gbps) data rates.

### Application of 5G

The 5G will continue to be applied in areas where 4G and 3G are currently used. However, critical applications will benefit from network speed and capacity to handle many devices. To highlight a few, the 5G has many applications, including:

- (a) Healthcare –where doctors and patients will be able to stay more linked with 5G, for example, through the use of wearable gadgets
- (b) Agriculture –sensors placed across the farm will alert the farmer to various scenarios. Also, wearable gadgets that monitor livestock health will assist owners in knowing their health status early.
- (c) Logistics - it will be easy and fast to track the shipping of items across multiple nodes.

### Exercise 7.3

Answer the following questions:

1. Analyse the trade-offs between applications deployed in the cloud and those installed on local premises.
2. Study carefully a scenario described in the section of the Internet of Things (IoT) and answer the following questions:
  - (a) Identify which physical processes were involved.
  - (b) Identify which cyber systems were involved.
  - (c) With a sketch diagram, explain how data would flow among all entities involved.
  - (d) Which types of cloud computing services would be appropriate in this scenario? Justify your answer.
3. Identify areas of 4IR applications that require a very high-speed network to function and demonstrate how 5G may be used.

### Impact of emerging technologies

Emerging digital technologies that drive 4IR have impacted various fields, and more are expected. This section looks at the impact of the fourth industrial revolution that results from emerging technology. This impact is on employment, skills and nature of work, business, economy, security, society, and individuals.

## Impact on employment DO NOT DRAFT

The fourth industrial revolution creates fewer jobs in new industries than in previous revolutions. That is especially true in light of rising levels of automation in industry, as human occupations are increasingly being replaced by automated machinery. For example, the automation of some work in offices and manufacturing industries, results in fewer jobs. Furthermore, in ride-sharing, featured by seamless algorithms, people, do not oversee the selection of which driver to meet and which customer to choose.

Emerging technologies will continue to alter the nature of employment in all sectors and vocations, and automation will increasingly replace human labour. Many diverse types of jobs have already been automated, particularly those that need mechanically repetitive and exact physical work. Many more will follow as computational power expands rapidly. This implies that different professionals must be aware of this trend and prepare themselves by being more innovative to remain relevant.

## Impact on skills and nature of work

Robotics and artificial intelligence are transforming how we live and work. Soon, other technologies such as machine learning, autonomous vehicles, and the Internet of Things will also change our lives. Building on the example of the postal service for sending and receiving mail, you can imagine how administrative processes in offices and the lives of individuals are changing. Previously mail carriers and individuals could walk to the post office to send or pick up mail. This manner of operation has been replaced by emails,

Short Message Service (SMS), and social media. As robotics and artificial intelligence technologies keep growing, systems that can send and receive emails are being invented. Regarding the nature of work, the need for physical working spaces is decreasing. Some employees can now work from remote areas on different continents and countries or even their homes.

These developments mean that new job skills will be required to adapt to a changing working environment characterised by emerging technologies. Most emerging technologies such as AI, augmented reality, robots, and cloud computing, will require personnel who can use these technologies. Computer programming, coding, and project management skills will be more in demand than today. Critical thinking, problem-solving, judgment and decision-making, interpersonal communication, and emotional intelligence are other core competencies needed in this situation.

## Impact on business

The impact of emerging technologies on business is observed from the perspectives of access over ownership and data-enhanced products. These perspectives are also referred to as *new business models* under the fourth industrial revolution.

### Access over ownership

Access over ownership is a business strategy that allows customers to utilise a product without purchasing and owning it. Access over ownership is also described as a sharing economy because it involves parties sharing an asset in space or time. People are starting to refrain from owning some assets and opting to access those assets.

It is because the costs of purchasing and owning some assets are high, and meanwhile, some assets, such as drilling machines, are rarely used by owners.

Under the dimension of space, people may share, for example, unused space in a personal car. Imagine driving alone from Dar es Salaam to Dodoma in a car with five seats while you are occupying one seat only. You share unused (idle) space when you allow other people to ride with you in your car. Those passengers do not need to own the vehicle (asset) but need access to it. Other examples of ride-sharing are services offered by Uber, Bolt, Taxify and Bla Bla Car. Moreover, vacation renting of homes or rooms and car leasing services are forms of sharing through renting but in the dimension of time.

Digital platforms that mediate interactions between service providers and consumers are critical to the operational effectiveness of the access over ownership business model. Typically, the asset provider (supplier) registers on the digital platform and gives relevant information such as the description of the service they deliver and the price. The consumer also creates an account on the digital platform by entering their information. Following that, algorithms find the optimal match between provider and consumer based on many parameters such as proximity, price, and preference. Digital platforms like Uber, Bolt, and Taxify act as intermediaries between the driver and the passenger, while seamless algorithms perform their finest work behind the scenes.

### Product data enhancement

An application example from the retail

business might be used to elaborate the supply of additional information on a product. Imagine you are shopping at a supermarket for a blender to use at home in making juice, and you are unfamiliar with that sort of blender. It is possible that the product information written on the package is insufficient to describe the blender and how to use it. A Quick Response (QR) code may be helpful in this scenario. A QR code may be attached to the blender by the manufacturer, allowing you to scan the code with your smartphone (using a specific mobile app). On scanning, more details about how the blender was manufactured, how to operate it and its life span are displayed in various formats such as text, image and video. In this case, a product has been enhanced with data. Furthermore, augmented reality technology may be applied to improve the product in this particular case.

### Impact on security

The impact of emerging technologies arises from the perspectives of cyber warfare and autonomous warfare. Cyberwarfare can take a variety of forms, including attacks on financial infrastructure (bank and payment systems) and safety infrastructure systems (such as warning systems). Autonomous weapons, a combination of drone technology with artificial intelligence, can choose and engage targets with limited or without human interaction, based on predefined criteria. Drones which are flying robots, can also be used as weapons in combat while getting controlled remotely. The robot will play a role in the war in future wars and involve deploying military robots and AI-powered automated weaponry.

## Impact on society and individuals

Emerging technology has a variety of impacts on society and individuals, including socio-economic inequality and anti-government objectives. We are already seeing how technology and robotics are displacing human labour. Existing and potential employees' skill sets are restricted and tailored to the market's requirements. Individuals with higher innovation abilities, such as bringing new ideas, business models, and products to market, will be valued more than those with lower expertise. This upward tendency suggests that those with the necessary abilities will survive and outperform the low-labour skilled workforce.

Digital media has both beneficial and harmful effects. Digital (social) media favour linking individuals to form friendships in one-to-one networks and social groupings. This networking stretches across national borders, allowing people to have incredible personal experiences and feelings. Digital media, particularly social media, which non-state players dominate, may influence society with extreme agendas. Some of these agendas have malicious intentions such as misinforming the public and causing unrest.

Furthermore, continued advancements in artificial intelligence and robotics raise the question of whether or not we, as humans, will be able to govern intelligent machines. What if they one day outstrip our ability to manage them? Algorithms are being used to choose who should attend an interview, establish prices for items in online stores, provide health advice, determine who to date, where to travel, and which hotel to book, among other things. Do we now

trust algorithmic recommendations or do we accept the opinions of our friends and close relatives? Despite continuous technological advancements, we as humans still need a personal touch. However, the trend reveals that our social ties, once deeply rooted in physical contact, are beginning to drift apart as social exchanges migrate to social media, an aspect that needs a balance between the two.

### Exercise 7.4

Answer the following questions:

1. Explain with examples how emerging technologies have started to impact people's way of living, changing them from possessing assets to accessing assets
2. Describe how robotics and automation affect skills and employment.
3. Identify the social impact of emerging technologies.

### Chapter summary

1. Unlike earlier industrial revolutions, the Fourth Industrial Revolution (4IR) is progressing exponentially. This is the outcome of our intimately interconnected society, and that new technology opens us to previously unimaginable possibilities. The 4IR builds on digital transformation and brings together a variety of emerging technologies that are causing extraordinary paradigm changes in socioeconomic development.
2. Emerging technologies drive the 4IR. In particular, emerging digital technologies such as artificial

intelligence, machine learning, Blockchain, Cloud computing, and 3D/4D/5D printing account for the 4IR. Others are Big data, the Internet of Things, drones, virtual and augmented reality, autonomous vehicles, robotics and the 5G network. Some of these are technologies already being applied in the industry while others have use cases and prototypes. These advancements have brought broad impacts on employment, skills and nature of work, business, economy, security, society, and individuals.

3. The impacts of emerging technologies are observed in employment, skills and the nature of work. In ride-sharing, for example, seamless algorithms, not people (employees), oversee the selection of which driver to meet and which customer to choose. Regarding skills and the nature of work, new job skills will be required to adapt to a changing working environment. For example, computer programming, coding, and project management skills will be more in demand than today.
4. In the business world, new business models such as access over ownership and product data enhancement, are being driven by rising digital technologies. Customers utilise a product (asset) without purchasing and owning but share (rent) a whole or part of that asset for the access over ownership business model. For product data enhancement, items may now communicate about their current states, and other products can now provide extra information about themselves.

5. Other impacts are on security and society. Cyberwarfare attacks threaten financial infrastructure (bank and payment systems) security. Based on pre-defined criteria, autonomous weapons that combine drone technology with artificial intelligence are being engaged to hit targets with limited or without human interaction. In a social context, emerging technologies are causing socio-economic inequality and anti-government objectives.

### Revision exercise

Fill in the blanks

1. The \_\_\_\_\_ uses your existing real environment and overlays virtual information on top of it.
2. \_\_\_\_\_ learns from data to perform functions that are similar to those performed by people.
3. \_\_\_\_\_ learns from humans to perform functions that are similar to those performed by people.
4. \_\_\_\_\_ is made up of data that cannot be changed.
5. Discuss with examples the benefits and drawbacks of cloud data storage for businesses
6. Examine how law enforcement officials, particularly police officers, can use wearable computers in their daily duties.

## FOR ONLINE USE ONLY

7. Describe the benefits of using QR code attached to product packages to:
- Manufacturers
  - Retailers
8. Explain the difference between augmented reality and virtual reality.
9. A vast amount of data about academic records exists at a particular high school in Tanzania. The high school intends to safeguard such information by regularly backing up its records. The school's ICT department is contemplating using one of the following backup systems; tape, hard drive, or cloud storage. Evaluate the three options and recommend, with reasons, the most suitable one.
10. How can wearable devices improve health care?
11. Discuss the impact of social media in business.
12. What is the influence of the fourth industrial revolution on our daily lives?
13. Give five (5) examples of things you can print by using 3D technology.
14. Compare 3D-printed products and conventionally manufactured goods in terms of strength and quality.
15. Emerging technologies have both positive and negative impacts on society globally. Discuss four of them.
16. Consider any four emerging technologies. Discuss their impact on people in Tanzania now and in the future.

FOR ONLINE USE ONLY  
DO NOT DUPLICATE

## Glossary

|                    |                                                                                                                                                                                                                                                                                                                               |
|--------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 2D-Array           | It is an array containing a collection of cells laid out in a two-dimensional grid, like a table with rows and columns, even though the values are stored in linear in the memory                                                                                                                                             |
| 3-D Array          | It is a multi-dimensional array which is specified using three subscripts, block size, row size and column size. The more the dimensions in an array mean more data can be stored in that array. In short, it is an array of arrays.                                                                                          |
| A function         | It is a block of code that performs a specific task.                                                                                                                                                                                                                                                                          |
| A node             | It is a structure that contains a key or value and pointers in its child node in the tree data structure.                                                                                                                                                                                                                     |
| Abstract data type | It is a special kind of the data objects that make up a data type and the functions that operate on these objects. These are special kinds of the datatype whose behaviour is defined by a set of values and operations. The keyword “Abstract” uses these data types and performs different operations.                      |
| Access control     | It refers to a selective restriction of access to a place or other resource.                                                                                                                                                                                                                                                  |
| Algorithm          | It is a process or set of rules to be followed in calculations or other problem-solving operations, especially by a computer. It is also defined as a predetermined set of instructions for solving a specific problem in a limited number of steps.                                                                          |
| Array              | It is a data structure that can store a fixed-size collection of elements of the same data type. These elements are all of the same data type, such as a string or an integer. An array is a group of related data values (called elements) that are grouped together.                                                        |
| Attributes         | These describe characteristics or properties that define all items about a specific category applied to all column cells.                                                                                                                                                                                                     |
| Bubble sort        | Bubble sort is an algorithm which arranges a string of numbers or other elements in the correct order. The method works by inspecting each adjacent element in the string, from left to right, switching their positions if they are out of order until it is arranged in the required order (ascending or descending) order. |

|                       |                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|-----------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Candidate key         | It is a specific type of field in a relational database that can identify each unique record independently of any other data. A candidate key is an attribute or set of attributes that can uniquely identify a tuple. The remaining attributes in a table are candidate keys except for the primary key.                                                                                                                                               |
| Class                 | A class is used in object-oriented programming to describe one or more objects. It is a template for creating or instantiating specific objects within a program.                                                                                                                                                                                                                                                                                       |
| Composite key         | A composite key is a combination of two or more columns in a table that allows us to identify each row of the table uniquely. It is a type of candidate key formed by more than one column.                                                                                                                                                                                                                                                             |
| Conceptual data model | It is a Data Model that defines <b>WHAT</b> the system contains. Business stakeholders and Data Architects usually create this model. The purpose is to organise, scope and illustrate business concepts and rules.                                                                                                                                                                                                                                     |
| Fabrication           | It is the attacks which involve generating data, processes, communications, or other similar activities with a system.                                                                                                                                                                                                                                                                                                                                  |
| Foreign key           | Foreign keys are the column of the table used to point to the primary key of another table.                                                                                                                                                                                                                                                                                                                                                             |
| Interception attack   | An interception is an act when an unauthorised individual gains access to confidential or private information. Interception attacks are attacks against the network, the confidentiality objective of the CIA Triad.                                                                                                                                                                                                                                    |
| Interruption attack   | Refers to making a service degraded or unavailable for legitimate use. It is an attack against the availability of the network or service.                                                                                                                                                                                                                                                                                                              |
| Logical data model    | It defines the organisation of a set of data by standardising the people, places, things (entities) and the rules associated with them and how the system should be implemented. It is geared toward creating a technical map of data structures, rules, and relationships between them using standard language and notation. Furthermore, it provides all the information about the various entities and the relationships between them in a database. |

**FOR ONLINE USE ONLY  
DO NOT DUPLICATE**

|                     |                                                                                                                                                                                                                                                                                                                                                                                         |
|---------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Memory address      | It is an exact assigned location in RAM used to track where information is stored. The memory address is the location where the variable is stored on the computer. When the value is assigned to the variable, it is stored in the memory address.                                                                                                                                     |
| Merge sort          | A sorting technique that arranges data by continuously combining items in the list. Every single item in the original unordered list is combined with another, creating groups of two. Every two-item group is merged, creating groups of four until there is one ordered list.                                                                                                         |
| Modification attack | It is an attack against the integrity of the information by either changing, inserting or deleting a piece of information to or from the existing or original.                                                                                                                                                                                                                          |
| Physical data model | A Data Model that defines how the system will be implemented using a particular DBMS system. The goal is to get the database up and to run. It also helps visualise database structure by replicating database column keys, constraints, indexes, and other RDBMS features.                                                                                                             |
| Primary Key         | It is the first key used to identify one and only one instance of an entity uniquely in the table.                                                                                                                                                                                                                                                                                      |
| Recursive algorithm | It is an algorithm that calls itself with “smaller or simpler” input values and obtains the result for the current input by applying simple operations to the returned value for the smaller or simpler input.                                                                                                                                                                          |
| Relational database | A relational database organises data into rows and columns, collectively forming a table. Data is typically structured across multiple tables, which can be combined via a primary or foreign key. These unique identifiers demonstrate the different relationships which exist between tables, and these relationships are usually illustrated through different types of data models. |
| Risk management     | The process of identifying, assessing and controlling threats to an organisation’s capital and earnings.                                                                                                                                                                                                                                                                                |
| Selection sort      | It is a sorting algorithm that selects the smallest element from an unsorted list in each iteration and places that element at the beginning of the unsorted list until the required order is attained.                                                                                                                                                                                 |

**FOR ONLINE USE ONLY  
DO NOT DUPLICATE**

|                       |                                                                                                                                                                                                                                                                      |
|-----------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Stack</b>          | It is a linear data structure that follows the principle of Last In First Out (LIFO). This means the last element inserted inside the stack is removed first. A good example is a pile of plates.                                                                    |
| <b>Syntax</b>         | It is a set of rules which define what the various combinations of symbols mean. The syntax tells the computer how to read the code. It is also a very specific set of words with a specific order guiding how to write code when we give the computer instructions. |
| <b>Tree</b>           | In the data structure, a tree is a nonlinear hierarchical data structure and comprises a collection of entities known as nodes. It connects each node in the tree data structure using directed and undirected edges.                                                |
| <b>Tree topology</b>  | In networks, it is a type of network layout resembling a tree, with a root node at the top and leaves at the bottom.                                                                                                                                                 |
| <b>Tuple</b>          | It is a data structure consisting of multiple parts. (in a relational database) it is an ordered set of data constituting a record, a row that contains interrelated data about a particular entity                                                                  |
| <b>Typedef</b>        | It is a reserved keyword in the programming languages such as, C and C++. It is used to create an additional name for another data type, provide some meaningful names to the already existing variable in a particular program                                      |
| <b>Vulnerability</b>  | It is the weakness in an information system, system security procedures, internal controls, or implementation that could be exploited or triggered by a threat source. These are flaws in a computer system that weaken the overall security of the device/system    |
| <b>Amateur hacker</b> | Refers to unprofessional or inexperienced hacker                                                                                                                                                                                                                     |



## Bibliography

1. Beñat Bilbao-Osorio, Soumitra Dutta, and Bruno Lanvin (editors). (2013). The Global Information Technology Report 2013: Growth and Jobs in a Hyperconnected World. Geneva: World Economic Forum.
2. Chemwa, G and Mburu, S. (2016). Mastering computer studies. S. Mburu, G. *Longhorn Publisher LTD, Kenya*
3. Chemwa, G. and Mburu, S. (2005). Computer studies form 3, Longhorn Publishers (K) Ltd, Kampala, Uganda
4. Chen, V. (2022, June, 14). An Introductory Guide to Visual Programming. Bubble IO. <https://bubble.io/blog/visual-programming/>
5. Doorsamy, W., Paul, S. B., and Marwala, T. (2020). *The Disruptive Fourth Industrial Revolution: Technology, Society and Beyond*. Springer. <https://doi.org/10.1007/978-3-030-48230-5>
6. Farouzan, A. B(2013). Data communication and networking 5<sup>th</sup> Edition. McGraw Hill Education, USA
7. Gartner, Inc. (2022). Gartner Glossary. <https://www.gartner.com/en/information-technology/glossary>
8. Gnaneswari, G. (2020). Programming in Visual Basic 6.0, Bangalore University, India
9. Gupta, M. (2020). *Blockchain for dummies*. 3rd IBM Limited Edition, John Wiley & Sons, Inc., Hoboken, ISBN: 978-1-119-62196-6
10. Hallberg, B. (2015). Networking A beginner's guide 6<sup>Th</sup> edition. McGraw Hill Education, India
11. Halvorson, M (2010). Microsoft Visual Basic 2010, Microsoft Press, Redmond Washinton
12. Hodson, J. (2006). Local Area Networks 4<sup>th</sup> ed. Bookpower, UK.
13. ITL Education solution Limited. (2011). *Introduction to computer science 2<sup>nd</sup> edition*. Pearson, India
14. Kate T., Gauri S., Pooja Sehgal, Rupal Bansal, Nikhilesh Jasuja, Syed Ali Husnain Gilan (2022, Oct., 23). Hardware vs Software. Diffen.com. Diffen LLC, [https://www.diffen.com/difference/Hardware\\_vs\\_Software](https://www.diffen.com/difference/Hardware_vs_Software)

15. Konsbruck Robert Lee. (2002). *Impacts of Information Technology on Society in the new century*. *Business and Management*, 5(6), 46-55.
16. Kumar, V., & Sinha, D. (2021). A robust intelligent zero-day cyber-attack detection technique. *Complex & Intelligent Systems*, 7(5), 2211-2234.
17. McBride, P. K. (2002). *Introductory Visual Basic*. Thomson Learning, High House, 50-51 Bedford Row, London WC1 4LR
18. Mishra, S. (2015, December). Network security protocol for constrained resource devices in Internet of things. In 2015 Annual IEEE India Conference (INDICON) (pp. 1-6). IEEE.
19. Pal, S., Hitchens, M., Rabehaja, T., & Mukhopadhyay, S. (2020). Security requirements for the internet of things: A systematic approach. *Sensors*, 20(20), 5897.
20. Proudfoot, R. and Kelley, S. (2017). *Can technology save the planet?* WWF Australia.
21. Proudfoot, R., & Kelley, S. (2017). Can technology save the planet? WWF Australia. <https://www.wwf.org.au/ArticleDocuments/360/pub-can-technology-save-the-planet-30may17.pdf.aspx?Embed=Y>
22. Schwab, K. (2016). *The Fourth Industrial Revolution*. World Economic Forum. Cologny, ISBN-13: 978-1-944835-01-9
23. Schwab, K., Nadella, S., and Davis, N. (2018). *Shaping the Fourth Industrial Revolution*. World Economic Forum. ISBN 9781984822611, Ebook ISBN 9781984822628
24. Sensors, E. (2019). The Impact of Technology on the Environment and How Environmental Technology Could Save Our Planet. Accessed at: <https://edinburghsensors.com/news-and-events/impact-of-technology-on-the-environment-and-environmental-technology/>
25. Smith, A J (2011). *Visual Basic programs to accompany programming logic and design six edition*, Course Technology 20 channel centre street Boston, MA 02210 USA
26. UNACTAD. (2018). *Technology and Innovation Report: Harnessing Frontier Technology for Sustainable Development*. United Nations.
27. Writing, A. (n.a.). The Advantages & Disadvantages of Visual Basic. Techwalla. <https://www.techwalla.com/articles/the-disadvantages-of-ruby-programming>



- Access**, 285
- Access control**, 285, 286, 288, 322
- Active*, 285
- alternate keys, 87
- analog, 117, 118, 119, 146, 169, 172, 173, 185, 186, 189, 190, 193, 196
- Angry IP Scanner**, 308
- antenna*, 174, 175, 176, 177, 196, 359
- Arithmetic, 188
- ARP**, 155, 276, 309
- Array, 5, 6, 7, 8, 9, 10, 11, 12, 13, 15, 18, 24, 25, 26, 32, 45, 50, 52, 54, 57, 58, 59, 60, 61, 62
- artificial intelligence, 337, 345, 346, 365, 366, 368
- Artificial Intelligence, 324, 334, 337, 345, 365
- assessments*, 285
- Asynchronous, 119, 122, 125, 131, 190
- Attributes, 86
- Augmented Reality, 337, 360
- Authorization, 264, 285
- Base Band*, 172
- Batch, 74, 75
- Big data, 345, 357, 358, 368
- binary search, 34, 48, 49, 50, 60, 61
- Blockchain, 337, 339, 345, 347, 348, 349, 368, 372
- Bluetooth, 128, 129, 181
- Boolean**, 3, 4, 59
- bridge, 31, 62, 127, 138, 147, 148, 150, 182
- Broadband*, 172
- Brouter, 150
- Bubble sort, 54, 60
- Burp Suite**, 308
- Cain & Abel**, 308
- candidate keys, 87, 103, 104
- Character**, 4
- Classful, 160, 162, 164
- Classification, 2, 7, 119, 127, 128, 133,
- Clearing tracks**, 307, 308
- Cloud computing, 311, 324, 325, 326, 327, 350, 351, 368
- Cloud Developers*, 327
- Coaxial*, 127, 169, 171, 172, 173
- code, 12, 24, 38, 40, 56, 86, 107, 110, 273, 274, 277, 278, 294, 318, 320, 326, 328, 349, 350
- communication, 68, 71, 115, 116, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127, 128, 130, 131, 153, 154, 155, 156, 157, 159, 161, 168, 174, 176, 177, 178, 179, 180, 181, 185, 187, 188, 189, 190, 191, 192, 193, 196, 265, 275, 285, 287, 292, 294, 299, 322, 326, 331, 332, 336, 338, 341, 363, 365, 372

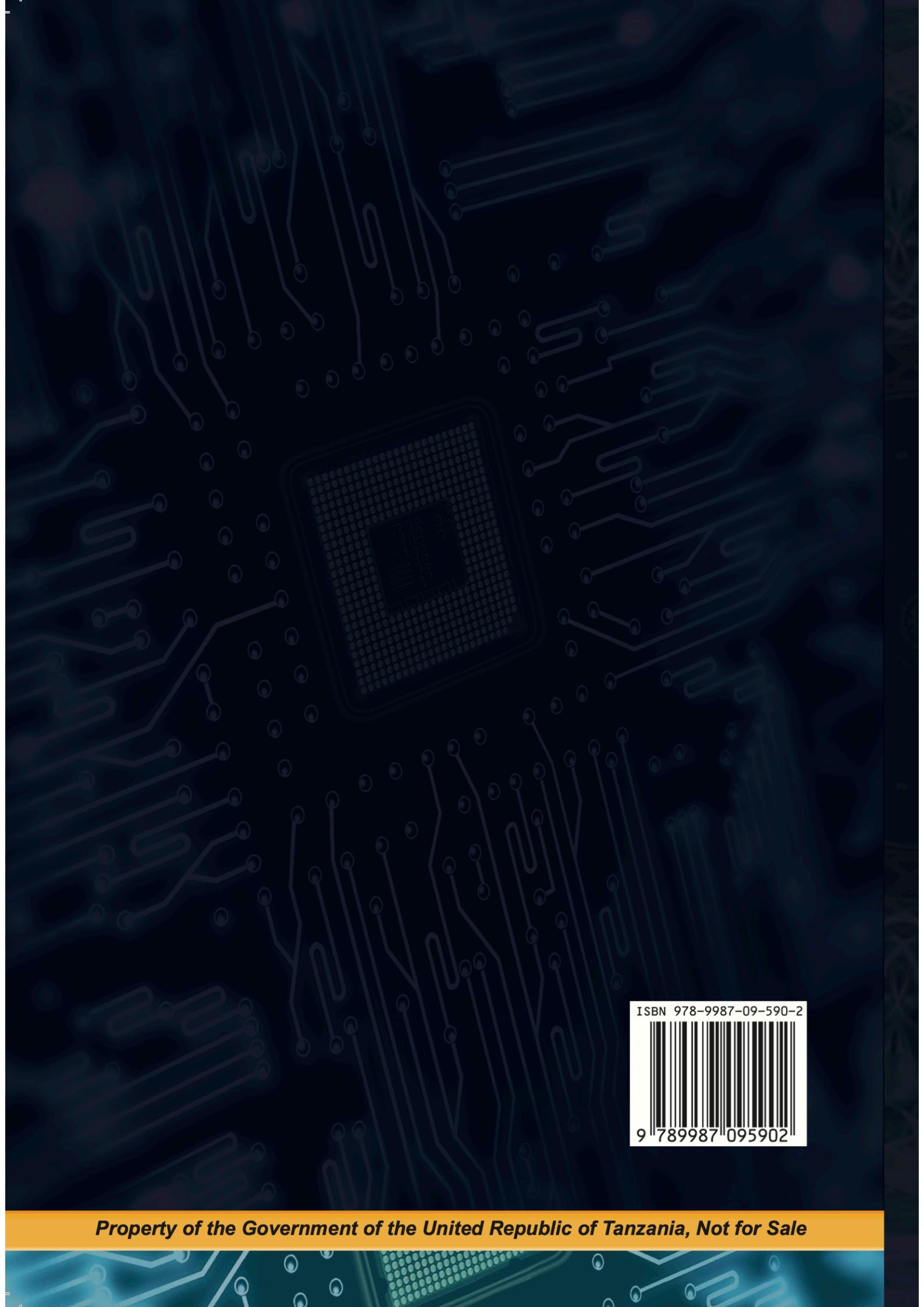
- composite, 16, 88, 101
- computer, 1, 2, 6, 12, 19, 38, 49, 59, 60, 64, 66, 67, 68, 69, 70, 72, 73, 75, 76, 77, 105, 115, 116, 121, 122, 127, 128, 129, 130, 131, 132, 134, 140, 141, 142, 143, 146, 150, 152, 154, 160, 161, 164, 168, 172, 184, 187, 192, 193, 194, 196, 262, 263, 264, 266, 267, 268, 269, 270, 271, 273, 275, 276, 279, 280, 281, 282, 284, 285, 289, 303, 304, 305, 308, 309, 310, 311, 313, 317, 318, 319, 320, 321, 322, 323, 325, 326, 332, 333, 334, 335, 337, 342, 345, 349, 350, 353, 360, 362, 368, 372
- computer programmer, 326
- Computer security, 262
- Concatenation**, 27, 28
- conquer, 50, 57, 58
- CPU, 31, 47, 302
- Crack -proofing**, 318
- Creation operation**, 27
- Cyber terrorism**, 305
- Data*, ix, 1, 2, 8, 11, 28, 50, 63, 69, 70, 75, 76, 78, 80, 82, 83, 91, 99, 101, 106, 107, 108, 110, 112, 115, 116, 117, 118, 119, 121, 122, 125, 129, 130, 144, 148, 150, 153, 155, 156, 157, 158, 159, 172, 179, 192, 193, 265, 266, 267, 268, 271, 285, 301, 302, 303, 311, 323, 324, 334, 351, 357, 363, 372
- Data Access**, 285
- data model, 79, 81, 82, 83, 84, 85, 92, 93, 100, 113, 117
- data structures*, 1, 2, 3, 4, 5, 6, 7, 14, 19, 24, 25, 32, 45, 59, 60, 82, 83, 99
- database, 17, 45, 63, 70, 74, 75, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 89, 90, 91, 92, 93, 94, 96, 98, 99, 100, 101, 105, 106, 107, 108, 110, 111, 112, 113, 262, 273, 277, 278, 282, 297, 299, 303, 305, 311, 326, 327, 347, 348, 349, 357
- DBMS, 78, 83, 84, 112
- decision, 33, 63, 64, 65, 68, 70, 74, 75, 76, 82, 150, 192, 295, 339, 345, 362, 363, 365
- Defence in depth**, 284
- demodulator, 146
- dependency, 82, 86, 96, 97, 98, 103, 104, 113
- depletion*, 161, 331
- digital library, 77
- Distinct type*, 88
- Do....while loop.*, 44
- domain, 85, 86, 87, 92, 101, 147, 150, 161, 279
- Double**, 4
- drones, 345, 358, 359, 360, 368
- Dynamic, 5, 24, 25, 149, 191
- emerging technologies*, 2, 326, 336, 337, 339, 364, 365, 366, 367, 368
- encapsulation*, 155, 157, 158, 159, 193, 196
- encryption, 112, 113, 268, 270, 285, 293, 302, 311

- Ethernet card**, 150, 151  
**Ethernet network**, 135, 180  
**Ettercap**, 308  
**Facial**, 346  
**firmware**, 262  
**foreign key**, 85, 87, 91, 92  
**Fragmentation**, 155  
frame, 100, 122, 123, 124, 134, 142, 148, 190, 294  
gadgets, 128, 333, 336, 364  
**Gateway**, 77, 149  
**Grey hat hackers**, 306  
**Hacktivist**, 306  
half-duplex, 126, 191, 192  
**Hard-disk loading**, 317  
hardware, 2, 37, 63, 68, 69, 70, 73, 83, 113, 115, 119, 120, 130, 138, 150, 152, 154, 168, 262, 263, 273, 280, 281, 296, 307, 308, 309, 310, 312, 313, 322, 325, 326, 332, 333, 334, 350  
HTTP, 153, 288, 290  
hub, 127, 129, 130, 135, 136, 147, 148, 194  
IaaS, 350  
immutable, 347, 348, 349  
Information, ix, 63, 64, 65, 67, 68, 74, 75, 76, 132, 144, 266, 267, 268, 270, 283, 312, 324, 325, 328, 330, 331, 334, 336, 372  
information system, 63, 64, 65, 66, 67, 68, 69, 72, 73, 74, 75, 76, 77, 113, 262, 265, 266, 283, 305, 322  
infrared, 127, 128, 129, 179, 180, 184, 193  
Inorder, 33, 34, 35  
**Insertion operation**, 27, 28  
Insertion-sort, 50, 52  
**Integer**, 4, 9  
Internet, vii, 68, 69, 70, 71, 115, 131, 132, 139, 153, 154, 155, 160, 161, 163, 164, 166, 180, 184, 193, 196, 267, 275, 277, 280, 284, 289, 291, 298, 299, 304, 311, 314, 316, 317, 326, 329, 330, 331, 332, 333, 334, 335, 337, 342, 345, 350, 351, 355, 356, 362, 363, 364, 365, 368  
**IP Address**, 160  
IPv4, 160, 161, 162, 299  
IPv6, 160, 161, 162, 196, 299  
iterative, 40, 44, 45, 60  
LAN, 71, 129, 130, 131, 155, 164, 168, 170, 173, 182, 183, 192, 277, 285  
LIFO, 28  
Linear data structure, 2, 4, 5, 28, 32, 60  
linear search, 46, 47, 48, 49, 50, 61  
Linear search, 46, 47, 60  
linked list, 5, 25, 26, 27, 28, 45, 59, 60, 61  
logical topology, 133  
**Mandatory**, 286  
Merge sort, 50, 58

- Metasploit**, 308
- Microwaves, 175, 176
- mobile, 62, 70, 77, 102, 128, 132, 178, 181, 268, 298, 322, 328, 329, 335, 336, 342, 347, 350, 357, 363, 366
- Modem, 146
- Multiplexer, 185, 186, 187, 188, 193
- Multiplexing*, 185, 186, 187, 188, 189, 190, 191, 193, 196
- MySQL, 77, 92, 106, 112
- Network, 70, 71, 80, 127, 128, 129, 130, 131, 132, 133, 135, 138, 140, 142, 144, 146, 149, 150, 151, 153, 154, 155, 158, 159, 160, 161, 163, 164, 165, 168, 180, 181, 182, 183, 184, 190, 193, 268, 280, 281, 283, 285, 289, 290, 291, 294, 295, 296, 297, 298, 307, 324, 325, 326, 327, 363
- network layer, 149, 150, 153, 154, 155, 160, 295, 300
- normalization, 91, 99, 101, 105, 113
- OSI, 151, 152, 153, 154, 155, 159, 193, 196
- PaaS, 350
- packets, 147, 148, 149, 150, 153, 155, 156, 157, 160, 161, 190, 276, 290, 292, 294, 298
- Passive*, 285
- Penetration testing**, 305, 309
- physical layer, 146, 152, 153, 193
- piracy**, 262, 316, 317, 318, 319, 322
- pollution*, 331
- POP, 28, 30, 153
- Postorder, 33, 36, 61
- Preorder, 33, 35, 37
- primary key, 85, 86, 87, 91, 92, 98, 102, 103
- Primitive, 2, 3, 4, 5, 27, 59, 60
- print, 32, 303, 353, 354, 355, 367
- Private Address*, 160, 164
- Programming*, ix, 1, 372
- project manager, 325, 327
- properties window
- protection*, 112, 119, 184, 262, 263, 265, 266, 267, 268, 270, 280, 283, 285, 304, 311, 314, 317, 318, 319, 321, 322, 351
- protocol, 147, 149, 153, 154, 155, 156, 157, 159, 160, 181, 192, 193, 197, 289, 290, 292, 294, 299, 300, 347
- PUSH, 28, 29
- query, 82, 89, 91, 109, 110, 111, 113, 114, 277, 278
- queue, 5, 26, 31, 32, 60, 61
- Queues, 31, 32
- Quicksort, 57, 59
- radio**, 70, 116, 127, 129, 175, 178, 179, 180, 182, 193
- real-time, 74, 75, 77, 115, 297, 361
- Records, 16, 17, 60
- recursion, 40, 41, 43, 57
- Recursive algorithms**, 40, 43

- recursive function, 40, 41, 43  
**Recycling**, 333  
**Red Hat Hackers**, 306  
 redundancy, 78, 80, 98, 101, 268, 282  
**Renting**, 317  
**Robustness**, 2  
 Router, 149, 150, 298  
**Script Kiddies**, 305, 306  
 security, 67, 91, 112, 113, 119, 132, 141, 169, 183, 190, 193, 196, 262, 263, 264, 265, 266, 267, 268, 269, 270, 271, 278, 280, 281, 282, 283, 284, 285, 286, 287, 288, 291, 293, 297, 298, 299, 300, 302, 303, 304, 305, 306, 307, 308, 309, 311, 312, 313, 314, 316, 321, 322, 323, 324, 325, 326, 327, 328, 330, 348, 350, 351, 356, 364, 366, 368  
 Selection-sort, 50  
 server, 106, 127, 130, 135, 140, 141, 142, 143, 144, 145, 193, 196, 197, 275, 276, 277, 278, 283, 284, 285, 294, 299, 303, 304, 311, 327, 343, 344, 347, 350, 351, 356  
 signal, 117, 118, 119, 122, 123, 125, 126, 133, 134, 146, 147, 168, 169, 172, 174, 175, 176, 177, 182, 183, 185, 186, 187, 188, 189, 190, 191, 192, 193, 363  
 Simplex, 119, 125, 126  
**Software**, 70, 131, 264, 316, 317, 318, 320, 322, 326, 350, 362  
**Software as a Service**, 350  
**Software counterfeiting**, 317  
**Sorting algorithm**, 50  
 stack, 5, 24, 28, 29, 30, 45, 61, 296  
 star, 133, 134, 135, 136, 147, 193, 197  
**Switch**, 148, 184  
 Syntax, 8, 42  
 system administrator, 164, 275, 305, 325, 326  
**Systems Analyst**, 325  
 TCP, 151, 154, 155, 156, 157, 159, 160, 161, 193, 196, 197, 290, 292, 296, 298  
 Telecommunication, 70  
 telephone, 64, 130, 132, 146, 169, 170, 172, 173, 174, 176, 187, 190, 191, 193, 341, 342  
 terminator, 133, 172  
 topology, 127, 132, 133, 134, 135, 136, 137, 138, 139, 147, 192, 197, 291, 307  
 transmission, 1, 70, 115, 116, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127, 131, 132, 133, 146, 148, 151, 153, 155, 168, 169, 170, 172, 173, 174, 175, 176, 177, 178, 179, 184, 186, 187, 188, 189, 190, 191, 192, 196, 197, 262, 267, 298, 322, 356  
**Transmitter**, 117  
 traversal, 26, 33, 34, 35, 36, 37  
**Traversing**, 27, 28  
 tree, 5, 32, 33, 34, 35, 36, 37, 60, 61, 79, 80, 133, 134, 138  
**Trojan**, 274, 280, 281, 309

- tuple, 85, 86, 87, 89  
Two-dimensional, 7, 8, 9, 80  
UDP, 153, 155, 156, 157, 161, 193,  
197, 290  
*Unmanned*, 337, 358, 359, 360  
*User Datagram Protocol*, 153, 155, 156  
user interface, 68, 329  
User-defined data, 4, 11, 19, 24  
variable, 4, 7, 11, 12, 14, 16, 26, 32, 60,  
161  
VB.Net  
VICOBA, 347, 348  
WAN, 71, 129, 131, 132, 155, 178, 182,  
192, 193  
Web, 132, 317, 324, 325, 328, 330  
*While loop*, 9, 44  
**White hat hackers**, 306  
Wi-Fi, 151, 180, 181, 182, 183, 193,  
276, 308, 309



ISBN 978-9987-09-590-2



**Property of the Government of the United Republic of Tanzania, Not for Sale**