



មហាវិទ្យាល័យវិស្វកម្ម  
FACULTY OF ENGINEERING

# Data Structure & Algorithm I

## Lecture 1

## Introduction to Data Structure & Algorithm

Chhoeum Vantha, Ph.D.

Telecom & Electronic Engineering

# Content

- DAS's Lecture organization
- Course objectives
- Introduction to DSA
- Programming Language
- Code Editor
- W1 – Lab1

# DAS's Lecture organization

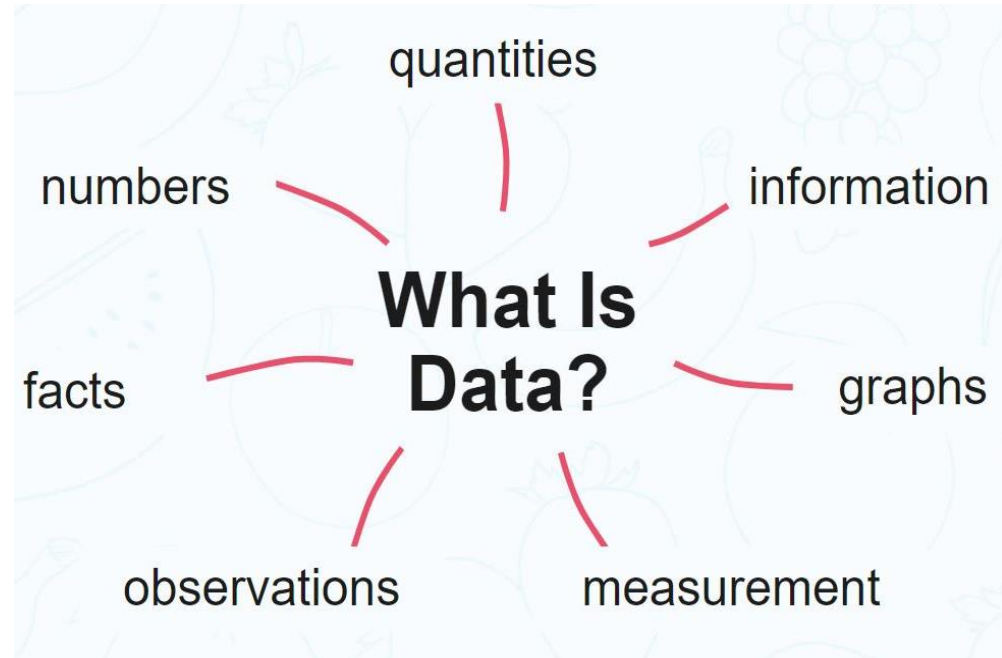
- Lecture 2 h
- Lab 2 h
  - Group of 5 students: 1 leader, 4 members
  - Randomly select to present at the end of the lab
  - Hand in the report at the end of the lab (Individule)
  - Quiz at the end of the lab to evaluate how you catch the lecture

# Course objectives

- Learn basic data structures and algorithms
  - Data structures – how data is organized
  - Algorithms – unambiguous sequence of steps to compute something
  - Algorithm analysis – determining how long an algorithm will take to solve a problem
- Become a better software developer
  - "Data Structures + Algorithms = Programs"
    - Niklaus Wirth, author of Pascal language

# What is Data?

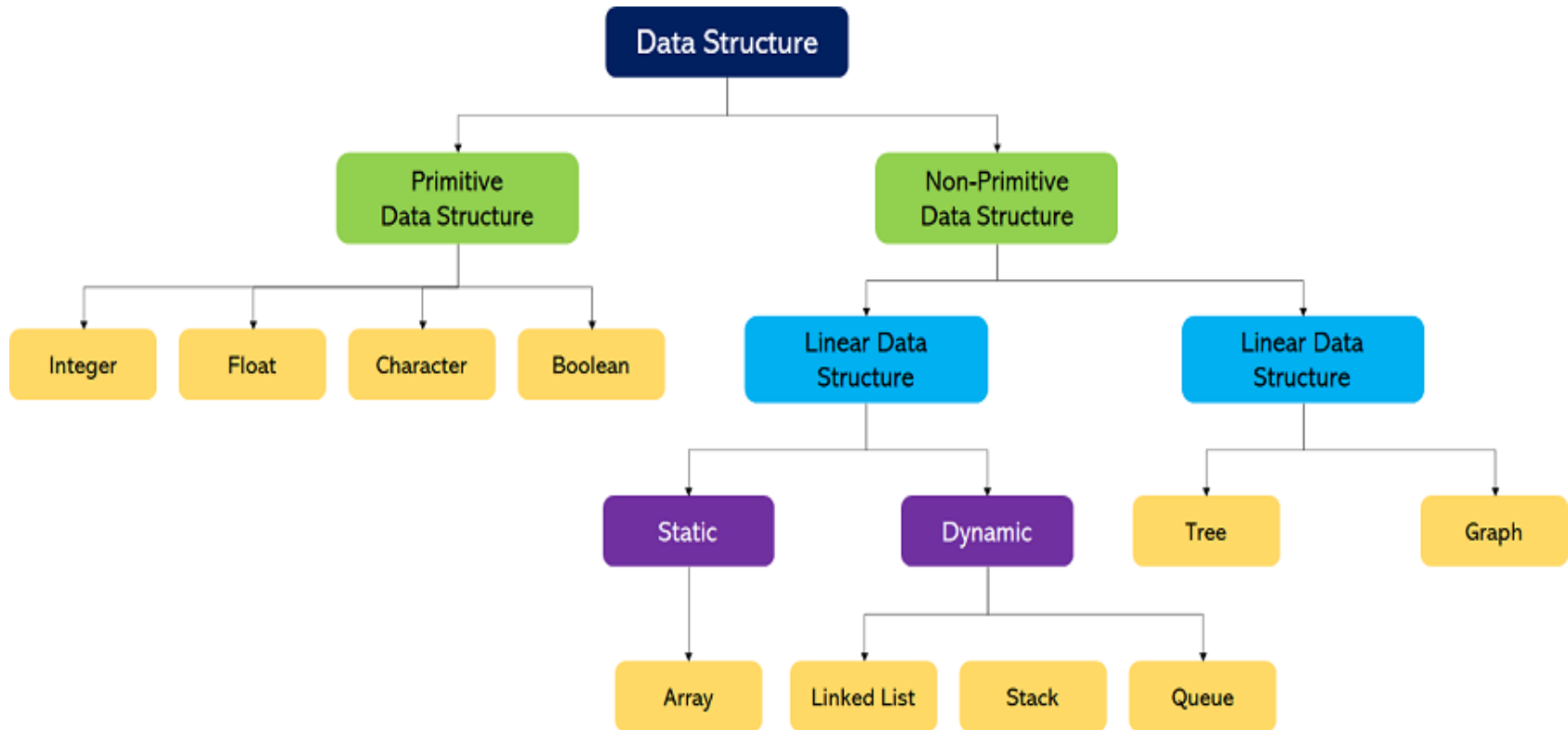
- Data is a collection of discrete or continuous values that convey information, describing the quantity, quality, fact, statistics,...



# What is Data Structure?

- **Data structure** is a representation of data and the operations allowed on that data.
- A data structure is a way to **store** and **organize data** to **facilitate access** and **modifications**.
- Data Structure is the method of representing logical relationships between **individual data elements** related to the **solution of a given problem**.

# Types of Data Structure



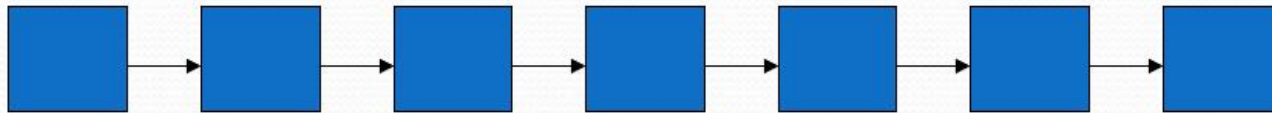
**Primitive data Structure:** It is a primitive data type, that can hold a single value.

**Non-primitive data structure:** Data structure that can hold more than one value

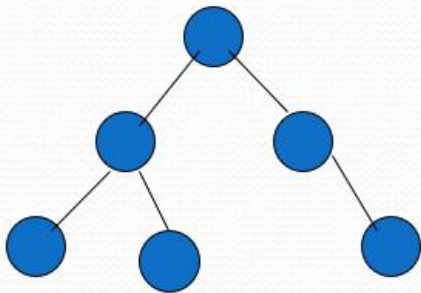
# Types of Data Structure



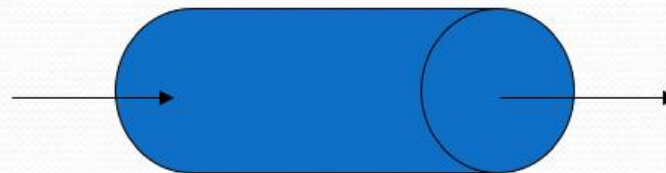
array



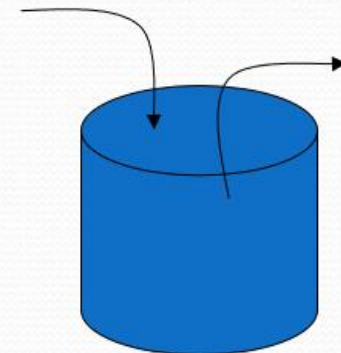
Linked list



tree



queue



stack



# Types of Data Structure

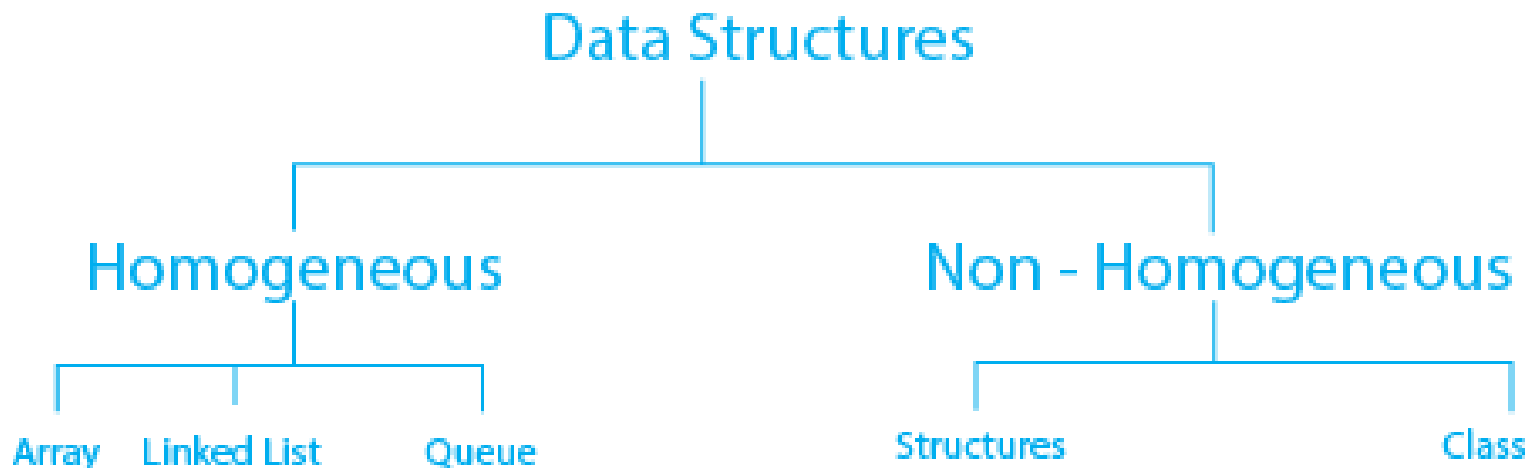
- **Linear:** In Linear data structure, values are arranged in linear fashion.
  - **Array:** Fixed-size
  - **Linked list:** Variable-size
  - **Stack:** Add to the top and remove from top
  - **Queue:** Add to back and remove from front
  - **Priority queue:** Add anywhere, remove the highest priority

# Types of Data Structure

- **Non-Linear:** The data values in this structure are not arranged in order.
  - **Hash tables:** Unordered lists which use a 'hash function' to insert and search
  - **Tree:** Data is organized in branches.
  - **Graph:** A more general branching structure, with less strict connection conditions than for a tree

# Data Structure - Classified

- **Homogenous:** consist of all the elements of similar data types
- **Non-homogenous:** must not consist of all the elements of similar data type



# The Core Operations of ADT

- Every Collection ADT should provide a way to:
  - **add** an item
  - **remove** an item
  - **find**, **retrieve**, or **access** an item
- Many, many more possibilities
  - is the collection empty
  - make the collection empty
  - give me a subset of the collection

# Selection of Data Structure

- It must be **rich enough in structure** to represent the relationship between data elements
- The structure should be **simple enough** that one can effectively process the data when necessary

# Programming Language

What is the **Programming Language**? Name?



# Programming Language

- Top 10 Most Popular Programming Languages:

1) Python

2) JavaScript

3) Java

4) C#

5) C

**6) C++**

7) Go

8) R

9) Swift

10) PHP

# Code Editor

What is a Code editor? Name??





# Code Editor

- Top 10 Best FREE Code Editor & Coding Software:

1) UltraEdit

2) Atom

3) Sublime Text

4) Notepad++

5) Brackets

**6) Visual Studio Code**

7) Vim

8) Bluefish

9) TextMate

10) TextWrangler

# W1 – Lab 1

# Getting To Start with C++

- Install **C++ programming** with **Visual Studio Code**

# Getting To Start with C++

- Basic Structure of a C++ program
- Basic Data Type using C++
- Control Statement in C++
- Structure of Function and Class in C++

# Basic Structure of a C++ program

Describe the C++ program below:

```
1  #include <iostream>
2  using namespace std;
3
4  int main() {
5      cout << "Hello World!" << endl;
6      return 0;
7  }
```

# Basic Structure of a C++ program

Describe the C++ program below

- Standard libraries section
- Main function section
- Function body section

```
1  #include <iostream>
2  using namespace std;
```

# Basic Structure of a C++ program

Describe the C++ program below

- Standard libraries section
- Main function section
- Function body section

```
4 ✓int main() {  
7 }
```

# Basic Structure of a C++ program

Describe the C++ program below

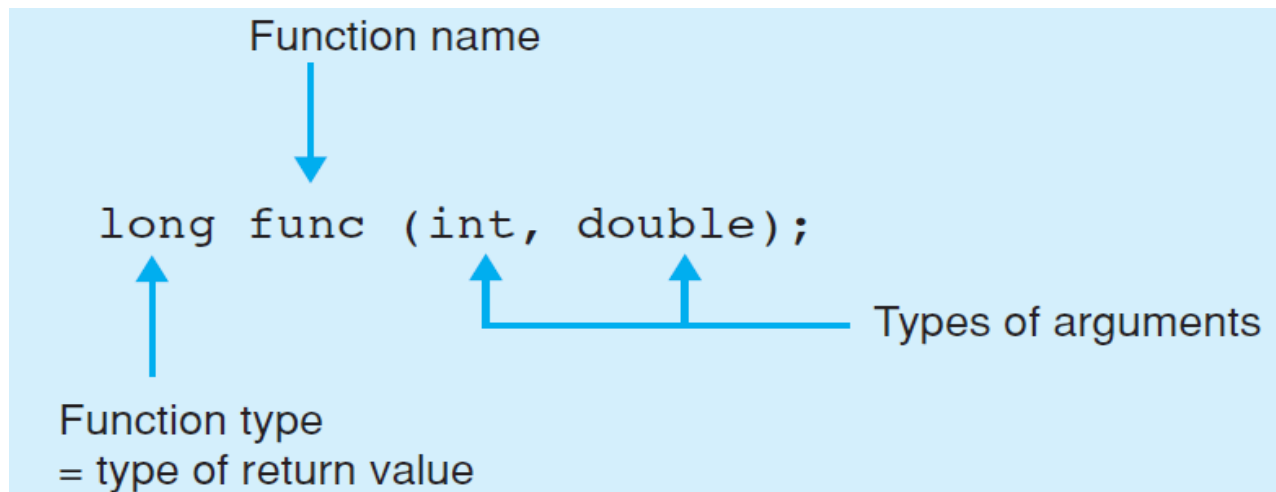
- Standard libraries section
- Main function section
- Function body section

```
5   cout << "Hello World!" << endl;  
6   return 0;
```



# Function - Declaration

The syntax to declare a function is:



```
returnType functionName (parameter1, parameter2,...) {  
    // function body  
}
```

```
// function declaration  
void greet() {  
    cout << "Hello World";  
}
```

## Ex 1

Capture your Code Editor with successful installation and place it in the word field.

## Ex 2

Write a program in C++ to print the multiplication table

Example: Input the number of multiplication: 3

=> Copy code to word field

1	×	3	=	3
2	×	3	=	6
3	×	3	=	9
4	×	3	=	12
5	×	3	=	15
6	×	3	=	18
7	×	3	=	21
8	×	3	=	24
9	×	3	=	27
10	×	3	=	30

## Ex 3

Write a program in C++ to check the status of any number whether it is even or odd number.

**Expect Output:**

```
Enter any Number: 15  
The number 15 is the ODD number
```

Thanks!