Academic Year: 2020

Semester **3**

Course Code: **CS-212L**

Course Title: **Data structures and Algorithms Lab**

CS212L-Data Structure and algorithms

Type of Lab: Open Ended **Weightage:**

CLO 1: CLO's.

State the Rubric	**Cognitive/Understanding**	CLO1	Rubric A

Rubric A: Cognitive Domain

Evaluation Method: GA shall evaluate the students for Question according to following rubrics.

CLO	0	1	2	3	4
CLO1	Mention Milestones with respect to rubrics				

CS212L-Data Structure and algorithms

Lab 2

Analysis of Algorithms with respect to Time complexity

Objectives: To improve and rebuilt the concepts of basic programming.

A Once Glance On Prerequisite Knowledge :

Variables and Data types

Variables:

A variable is a named memory location, container (storage area) to hold data. Each variable should be given a unique name (identifier)

Variable Declaration:

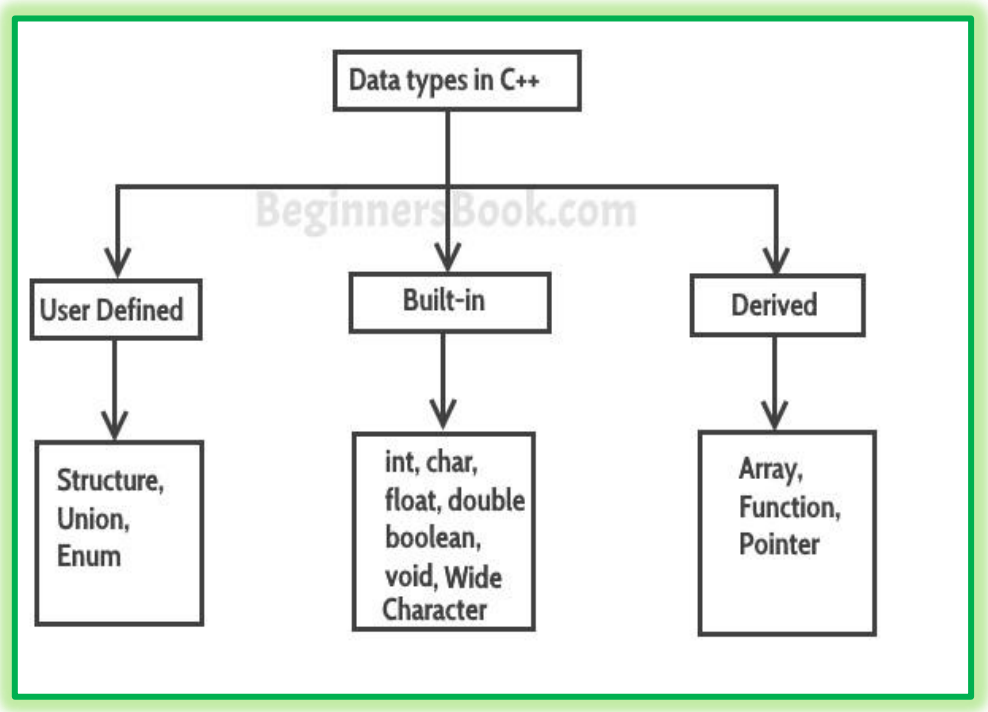
Syntax: **Data_Type Variable_name ;**

Variable Initialization:

Syntax: **Data_Type Variable_name = Value ;**

Data Types/Types of Variables:

1. **Primitive Data Types**:
2. **Derived Data Types:**
3. **Abstract or User-Defined Data Types:**



Data Type	Size	Description
int	4 bytes	Stores whole numbers, without decimals
float	4 bytes	Stores fractional numbers, containing one or more decimals. Sufficient for storing 7 decimal digits
double	8 bytes	Stores fractional numbers, containing one or more decimals. Sufficient for storing 15 decimal digits

boolean	**1 byte**	Stores true or false values
char	**1 byte**	Stores a single character/letter/number, or ASCII values

Operators in C++

Operators are used to perform operations on variables and values.

- **Arithmetic operators**
- **Assignment operators**
- **Comparison operators**
- **Logical operators**

C++ User Input

- You have already learned that cout is used to output (print) values. Now we will use cin to get user input.
- cin is a predefined variable that reads data from the keyboard with the extraction operator (>>).
- In the following example, the user can input a number, which is stored in the variable x. Then we print the value of x:

C++ Strings

- Strings are used for storing text set of characters.
- We must include an additional header file, the <string> library:

Example

```
// Include the string library
#include <string>
```

```
// Create a string variable
string greeting = "Hello";
```

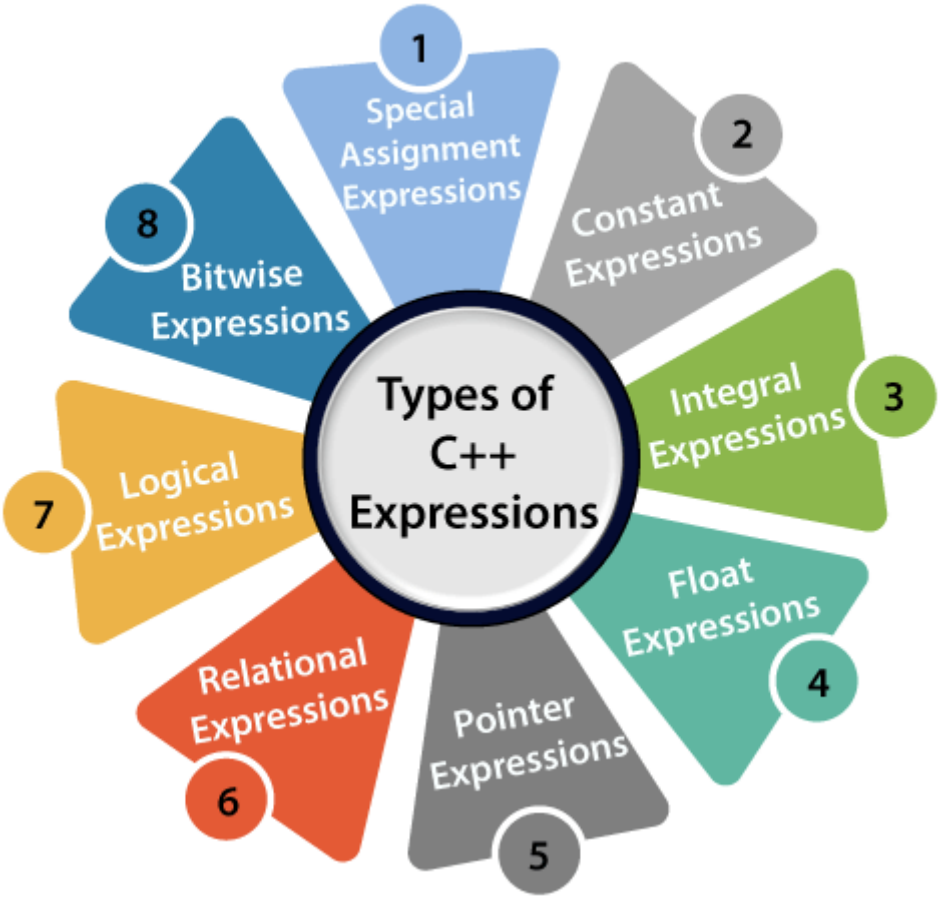
User Input Strings:

Different functions can be used to take string inputs:

1. Using cin
2. Using cin.getline()
3. Using cin.get()

C++ Expression

- C++ expression consists of operators, constants, or variables.
- An expression can consist of one or more operands, and one or more operators to compute a value.
- Every expression produces some value which is assigned to the variable with the help of an assignment operator.



Conditional Statements in C++

- Also known as selection statements, Decision making statement or control structures.
- Used to make decisions based on a given condition.
- If the condition evaluates to True, a set of statements is executed, otherwise another set of statements is executed.

Types of Conditional Statements

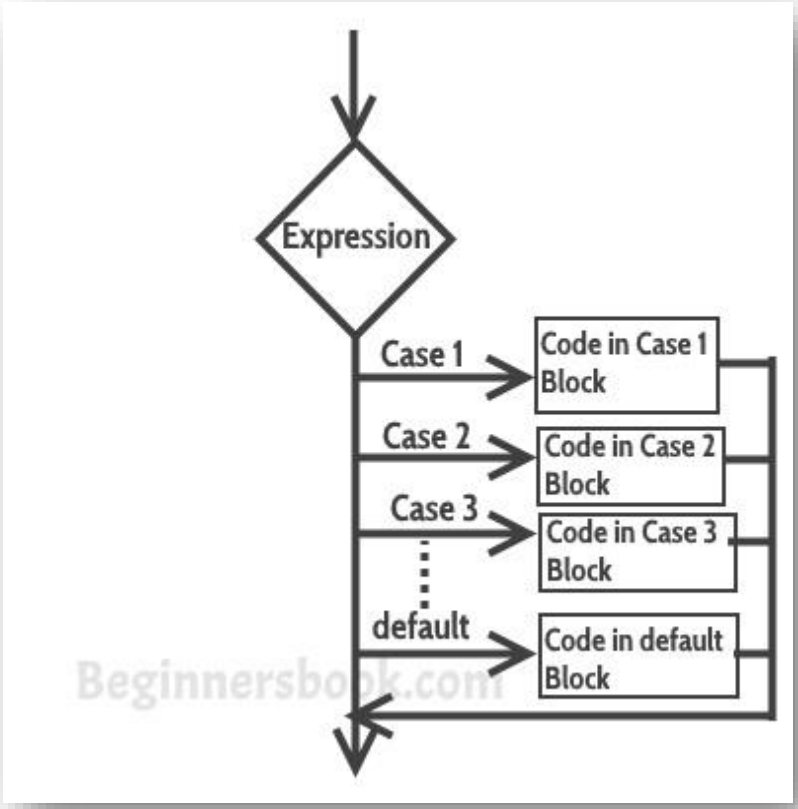
A. **if statement**

Syntax:

```
if(condition)
{
```

```
        //code to be executed
}
```

B. **if-else statement**
C. **nested if statement**
D. **if-else-if**
E. **Switch statement**

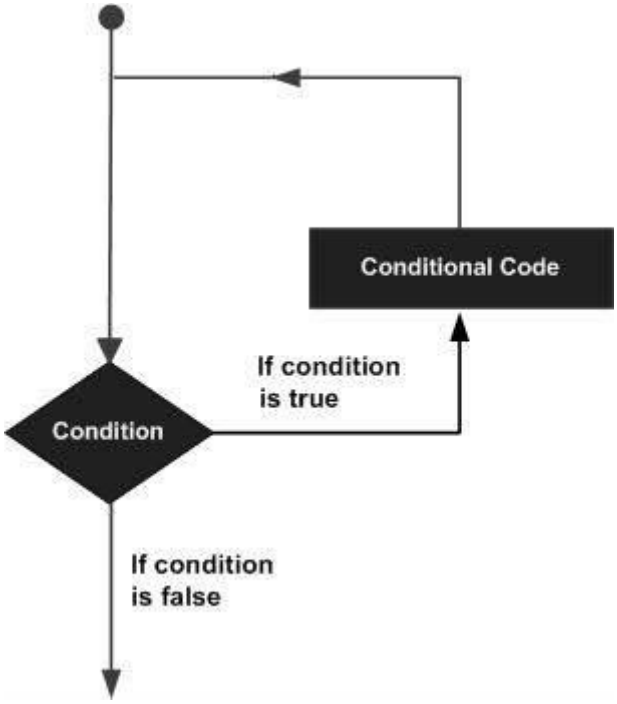


Loops in C++

There may be a situation, when you need to execute a block of code several number of times. In general, statements are executed sequentially: The first statement in a function is executed first, followed by the second, and so on.

Programming languages provide various control structures that allow for more complicated execution paths.

A loop statement allows us to execute a statement or group of statements multiple times and following is the general from of a loop statement in most of the programming languages −



C++ programming language provides the following type of loops to handle looping requirements.

Sr.No	Loop Type & Description
1	**while loop** Repeats a statement or group of statements while a given condition is true. It tests the condition before executing the loop body.
2	**for loop** Execute a sequence of statements multiple times and abbreviates the code that manages the loop variable.
3	**do...while loop** Like a 'while' statement, except that it tests the condition at the end of the loop body.

4	**nested loops** You can use one or more loop inside any another 'while', 'for' or 'do.while' loop.

To learn more about loops:

https://www.tutorialspoint.com/cplusplus/cpp_loop_types.htm

Arrays in C++

In C++, an array is a variable that can store multiple values of the same type.

For example,

Suppose a class has 27 students, and we need to store the grades of all of them.

Instead of creating 27 separate variables, we can simply create an array:

```
double grade[27];
```

Here, `grade` is an array that can hold a maximum of 27 elements of `double` type.

In C++, the size and type of arrays cannot be changed after its declaration.

C++ Array Declaration

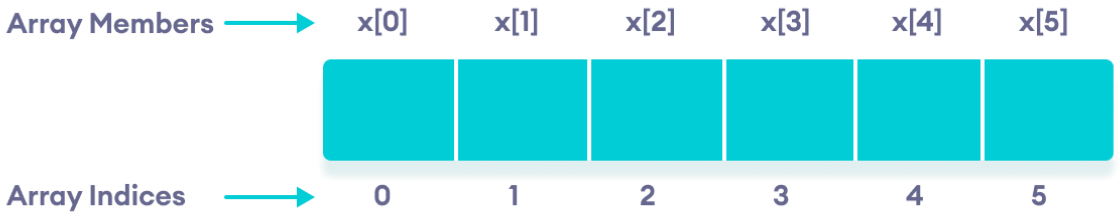
```
dataType arrayName[arraySize];
```

For example,

```
int x[6];
```

Here,

- `int` - type of element to be stored
- `x` - name of the array
- `6` - size of the array



To learn more about arrays: https://www.programiz.com/cpp-programming/arrays

Functions in C++

- A function is a block of code which only runs when it is called.
- You can pass data, known as parameters, into a function.
- Functions are used to perform certain actions, and they are important for reusing code: Define the code once, and use it many times.
- C++ provides some pre-defined functions, such as main(), which is used to execute code. But you can also create your own functions to perform certain actions.
- To create (often referred to as *declare*) a function, specify the name of the function, followed by parentheses **()**:

Syntax

```
void myFunction() {
  // code to be executed
}
```

Example Explained

- myFunction() is the name of the function

- void means that the function does not have a return value. You will learn more about return values later in the next chapter
- inside the function (the body), add code that defines what the function should do

Function Declaration and Definition

A C++ function consist of two parts:

- **Declaration:** the function's name, return type, and parameters (if any)
- **Definition:** the body of the function (code to be executed)

```
void myFunction() { // declaration
  // the body of the function (definition)
}
```

Note: If a user-defined function, such as myFunction() is declared after the main() function, **an error will occur**. It is because C++ works from top to bottom; which means that if the function is not declared above main(), the program is unaware of it:

Example

```
int main ()

{
  myFunction();
  return 0;
}
void myFunction()

{
  cout << "I just got executed!";
}
// Error
```

To learn more about functions visit:
https://www.w3schools.com/cpp/cpp_functions.asp

Class activity

1. Write a program to find out the area of triangle when three sides a, b and c of the triangle are given. Use appropriate statements to input the values of a, b and c from the keyboard. Formula for the area of triangle is **area = $(s(s-a)(s-b)(s-c))^{1/2}$** where **s=(a+b+c)/2**
2. Write a program to print the ASCII value of a character.
3. Write a program that inputs two numbers in main() function, passes these numbers to a function. The function will display the maximum number.
    a. Paula and Danny want to plant evergreen trees along the back side of their yard. They do not want to have an excessive number of trees. Write a program that prompts the user to input the following:
        i. The length of the yard
        ii. The radius of the fully-grown tree
        iii. The required space between fully grown tree

    The program outputs the number of trees that can be planted in the yard and the total space that will be occupied by the fully-grown tree.
4. Write a program to check whether a number is prime number, even number or odd number using function.
5. Write C++ program to calculate the factorial of number and ask user to enter Y if he wants to try another number or N if he wants to terminate the program. For example, if user enters 4 then output of the program will be 4*3*2*1 = 24 and user will be asked to type "Y/N". If user enters Y, then you will again ask for 'n' and recalculate the factorial. You will keep on doing it until users enters 'N'. If user enters 'N', program should exit.
6. Write a C++ program which asks user to enter two numbers: a dividend (the top number in a division) and a divisor (the bottom number). It then calculates the quotient (the answer) and the remainder, using the / and % operators, and prints out the result also ask user to try again.(if user enter 'y' or 'Y' then program again ask for two number if user enter 'n' or 'N' then it terminate). Solve this problem using **for loop control structure.**

    Expected Output:

```
Enter dividend: 11
Enter divisor: 3
Quotient is 3, remainder is 2
Do another? (y/n): y
Enter dividend: 222
Enter divisor: 17
Quotient is 13, remainder is 1
Do another? (y/n): n
```

7. Write a program that uses four arrays **numbers, squares**, **cubes** and **sums** each consisting of 5 elements. The **numbers** array stores the values of its indexes, the **squares** array stores the squares of its indexes, the **cubes** array stores the cubes of tits indexes and **sums** array stores the sum of corresponding indexes of three arrays . The program should display the values of all arras and the total of all values in **sums** array.

    Output:

```
Numbers: 0   1   2   3   4
Squares: 0   1   4   9   16
Cubes:   0   1   8   27  64
Sums:    0   3   14  39  84
Grand total: 140
```

8. Write a program that initializes an array. It inputs a value from the user and searches the number in the array. Finally, it will display the location of number in array if it exists. (using sequential search)
9. Write a program that inputs two numbers and one arithmetic operator in main() function, passes them to a function. The function applies arithmetic operation on two numbers based on the operator entered by user using **switch** statement.
10. Write a program that inputs base and height of a triangle in main function and passes them to a function. The function finds the area of triangle and returns it to main function where it is displayed on the screen. **Area = 1/2(Base*Height)**
11. Write a program that calls a function for five times using loop. The function uses a static variable initialized to 0. Each time the function is called, the value of static variable is incremented by 1 and is displayed on the screen.

Tasks to be submitted:

1. Write a program to generate a Pascal's triangle using function as follows:

```
1
1 1
1 2 1
1 3 3 1
1 4 6 4 1
```

2. **(Fraction calculator)** Write a program that lets the user perform arithmetic operations on fractions. Fractions are of the form a/b, in which a and b are integers and b 6¼ 0. Your program must be menu driven, allowing the user to 482 | Chapter 8: User-Defined Simple Data Types, Namespaces, and the string Type 8 select the operation **(+, -, *, or /)** and input the numerator and denominator of each fraction. Furthermore, your program must consist of at least the following functions:

   a. Function menu: This function informs the user about the program's purpose, explains how to enter data, and allows the user to select the operation.

   b. Function addFractions: This function takes as input four integers representing the numerators and denominators of two fractions, adds the fractions, and returns the numerator and denominator of the result. (Notice that this function has a total of six parameters.)

   c. Function subtractFractions: This function takes as input four integers representing the numerators and denominators of two fractions, subtracts the fractions, and returns the numerator and denominator of the result. (Notice that this function has a total of six parameters.)

   d. Function multiplyFractions: This function takes as input four integers representing the numerators and denominators of two fractions, multiplies the fractions, and returns the numerators and denominators of the result. (Notice that this function has a total of six parameters.)

   e. Function divideFractions: This function takes as input four integers representing the numerators and denominators of two fractions, divides the fractions, and returns the numerator and denominator of the result.
   (Notice that this function has a total of six parameters.)
   Some sample outputs are:
   3 / 4 + 2 / 5 = 23 / 20
   2 / 3 * 3 / 5 = 6 / 15
   Your answer need not be in the lowest terms.
3. There are 11 hard disks. One of them is used for backup and rest for saving one digit. One hard disk gets corrupted. How will you retrieve loss data? Write a C++ program for this

problem. Ask about values of all the hard disks and use back up hard disk as storage for your algorithm. Ask user again about the number of one hard disk which gets corrupted and save -1 in it. Using your algorithm and recover the corrupted data. You should not duplicate data of your hard disks in any other variable. You should focus on the question "how can you recover the data using your algorithm"

4. Write a program that asks the user to type 10 integers of an array and an integer value V and an index value i between 0 and 9. The program must put the value V at the place i in the array, shifting each element right and dropping off the last element. The program must then write the final array. For Example, if user enter index: 4 and value 59 then

Initial array :

58	**24**	**13**	**15**	**63**	**9**	**8**	**81**	**1**	**78**

After updating :

58	**24**	**13**	**15**	**59**	**63**	**9**	**8**	**81**	**1**

4. Write a program that uses three arrays **Mango**, **Orange** and **Banana** to store the number of fruits purchased by customer. The program inputs the number of mangos, oranges and bananas to be purchased by customer and stores them in corresponding array. The program finally displays the total bill of ach customer according to the following prices:

Rs. 20 per mango
Rs. 10 per orange
Rs. 5 per banana
The output should appear as follows:

****

Customer No.	**Mangoes**	**Oranges**	**Bananas**	**Total Bill**

****

1	5	10	12	Rs.260

REFRENCES: