**University of Management and Technology, Lahore Campus**

**Lab- 03 Manual**

Lab Instructor: Riaz Ahmad

Department of Computer Science

Email: [riazahmad@umt.edu.pk](mailto:riazahmad@umt.edu.pk)

## **Table of Contents**

[Lab # 03 2](file:///D:\UMT\Tahir\section%20u\UMT%20Tasks\Lab%20Manual%2003.docx#_Toc121297261)

[3.1 Objective: 2](#_Toc121297262)

[3.2 Scope: 2](#_Toc121297263)

[3.3 Useful Concept: 2](#_Toc121297264)

[3.4 Examples: 3](#_Toc121297265)

[3.5 Lab Exercise 7](#_Toc121297266)

[3.6 Home Work 7](#_Toc121297267)

### Lab # 03

**Data Types**

### Objective:

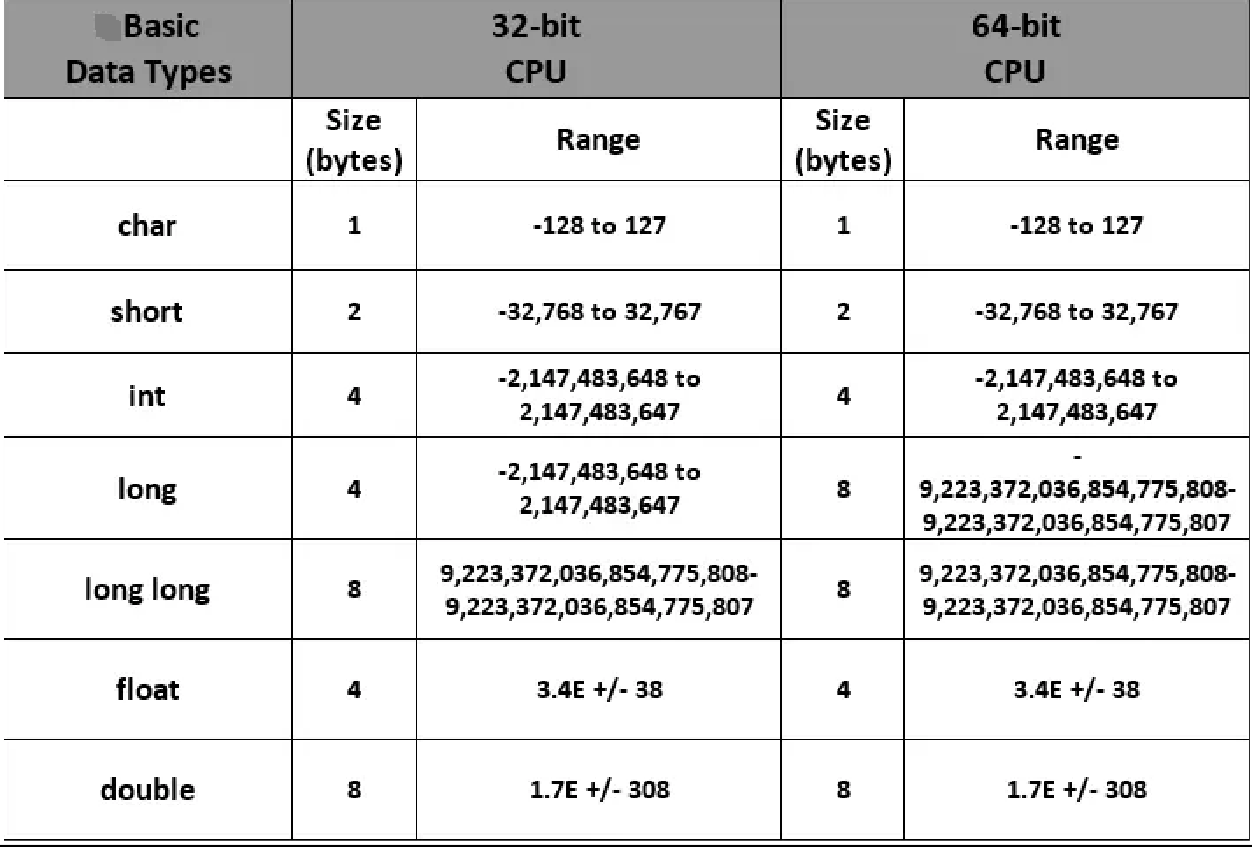
Learn the Problem Solving and Basics of C++ Language

### Scope:

The student should know the following:

* + - Problem Solving
    - Different data types of C and their Use.
    - Declaring Variables
    - Standard Input and Output
    - Writing Complete Programs

### Useful Concept:

A list of basic data types of C++, number of bytes used to store these data types in memory of computer system is:

**Variables**

* + - C++ variable is a named location in a memory where a program can manipulate the data. This location is used to hold the value of the variable.
    - The value of the C++ variable may get change in the program.
    - C++ variable might be belonging to any of the data type like *int*, *float*, *char* etc.

**Rules for naming C++ variable:**

1. Variable name must begin with letter or underscore.
2. Variables are case sensitive
3. They can be constructed with digits, letters.
4. No special symbols are allowed other than underscore.
5. sum, height, \_value are some examples for variable name

**Declaring & initializing C++ variable:**

* + - Variables should be declared in the C++ program before to use.
    - Memory space is not allocated for a variable while declaration. It happens only on variable definition.
    - Variable initialization means assigning a value to the variable.

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Type** | **Syntax** | **Example** |
| 1 | Variable declaration | data\_type variable\_name; | int x, y, z; char flat, ch; |
| 2 | Variable initialization | data\_type variable\_name = value; | int x = 50, y = 30; char flag = ‘x’, ch=’l’; |

### Examples:

**Example 3.1:** This example demonstrates declaration and initialization of variables

#include <iostream> using namespace std;

int main ()

{

// declaring variables: int a, b;

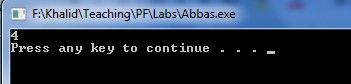
int result;

a = 5;

b = 2;

a = a + 1; result = a - b;

Here’s the program’s output:



cout << result; return 0;

}

**Example 3.2:** This program illustrates different techniques to initialize a variable.

// initialization of variables #include <iostream>

using namespace std;

int main ()

{

int a = 5; // initial value: 5

int b(3); // initial value: 3

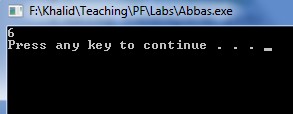
int c = 2; // initial value: 2. Also try int c{2};

int result; // initial value undetermined

a = a + b; result = a - c; cout << result;

return 0;

}

Here’s the program’s output:

**Example 3.3:** This program illustrates use of strings.

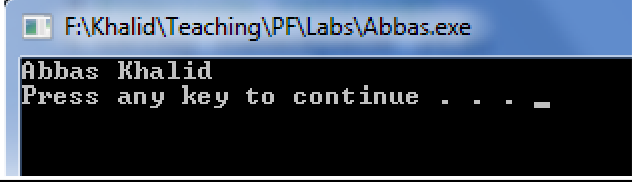
// my first string #include <iostream> #include <string> using namespace std;

int main ()

{

string myname;

Here’s the program’s output:

myname = " Abbas Khalid"; cout << myname;

return 0;

}

**Example 3.4:** This program calculates the area of the circle. The area of the circle is **π**r². The value of **π** is constant that is 3.14 but radius can change so this program gets the value of radius variable form user and calculate the area on that value.

#include <iostream> using namespace std;

int main ()

{

float radius,area;

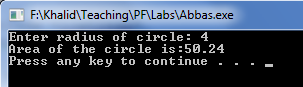
cout<<"Enter radius of circle: "; cin>>radius;

area = 3.14\*radius\*radius;

cout<<"Area of the circle is:"<< area<<endl;

return 0;

}

Here’s the program’s output:

**Example 3.5:** This program illustrates the addition on character values.

#include <iostream> using namespace std;

int main ()

{

char x, y ;

int z ; x = 'a' ;

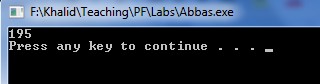
y = 'b' ;

z = x + y ; //Add the assci value of 'a' with assci value of 'b'. cout<<z<<endl;

return 0;

}

Here’s the program’s output:



**Example 3.6:** This program illustrates the use of sizeof() function which is used to find the memory space allocated for each C++ data type.

#include <iostream> #include <climits> using namespace std; int main ()

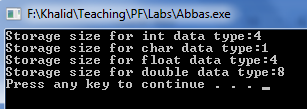
{

int a; char b; float c; double d;

cout<<"Storage size for int data type:"<<sizeof(a)<<endl; cout<<"Storage size for char data type:"<<sizeof(b)<<endl; cout<<"Storage size for float data type:"<<sizeof(c)<<endl; cout<<"Storage size for double data type:"<<sizeof(d)<<endl;

return 0;

}

Here’s the program’s output:

**Example 3.7:** A program to illustrate Octal, Decimal and Hexadecimal representation using ***cout***.

#include <iostream> using namespace std; int main()

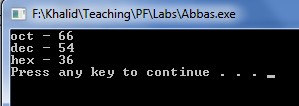
{

int n = 54;

cout << std::oct << "oct - " << n << '\n'; cout << std::dec << "dec - " << n << '\n'; cout << std::hex << "hex - " << n << '\n';

return 0;

}

Here’s the program’s output:

### Lab Exercise

**Exercise 3.1:** Write a program to compute circumference of a circle.

**Exercise 3.2:** Write a program that takes any ASCII value from user and display next five char after that ASCII value.

Hints: - if user enters 95, your program should display the char against the ASSCII value 96,97,98,99 and 100.

### Home Work

1. Write a program converts a temperature from Celsius to Fahrenheit. Use the following formula: F = 1.8 x C + 32.
2. Write a program that reads three integers representing hours, minutes, and seconds of a time. Then it should calculate the equivalent time in seconds.

|  |
| --- |
| #include <iostream>  #include <cmath>  using namespace std;  //2.    Write a program that reads three integers representing  // hours, minutes, and seconds of a time.  // Then it should calculate the equivalent time in seconds.  int main(){      int hours;      int minutes;      int seconds;      int totalSeconds;      cout << "Enter hours: ";      cin >> hours;      cout << "Enter minutes: ";      cin >> minutes;      cout << "Enter seconds: ";      cin >> seconds;      totalSeconds = ( hours \* 36000 ) + ( minutes \* 60 ) +      seconds ;      cout << "The total seconds are: " << totalSeconds;      return 0;  } |

1. Write a program that input length and width from the user, it calculates and displays area of rectangle by using formula Area=(length)(width).