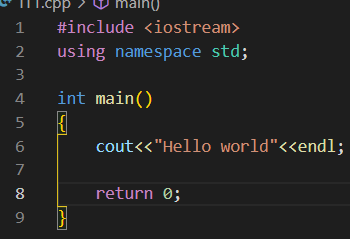
**LAB -1**

cin/cout basics, structures and classes

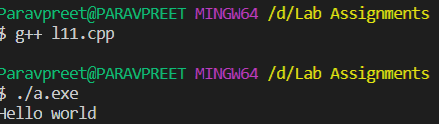
**Programming Question**

1. Write a program (WAP) to display "Hello World" on console display.

Code:

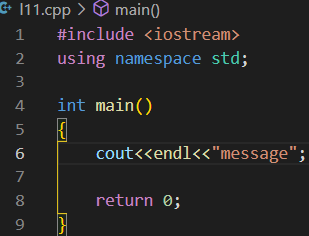


Output:

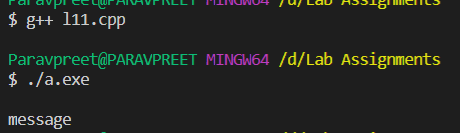


1. WAP to implement the following control characters:
2. ‘\n’ is for new line, or you can use *endl* – cout<<endl<<“message”;

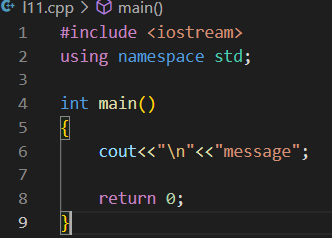
Code using endl:



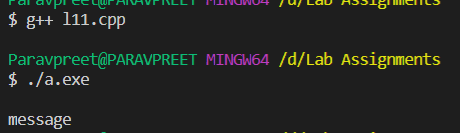
Ouput:



Code using \n:

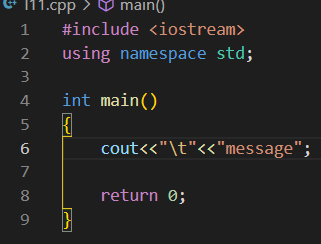


Output:

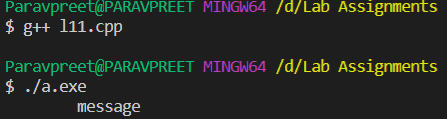


1. ‘\t’ is for tab

Code:

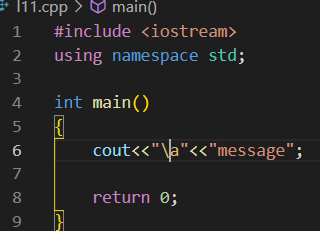


Output:

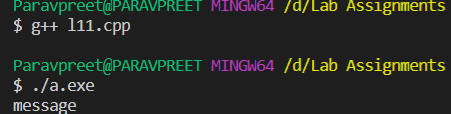


1. ‘\a’ is an alarm sound

Code:

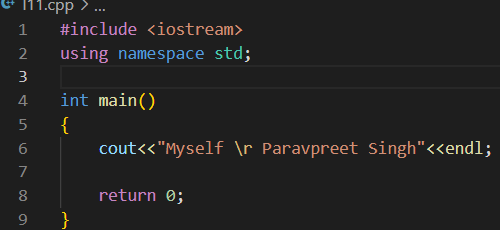


Output:



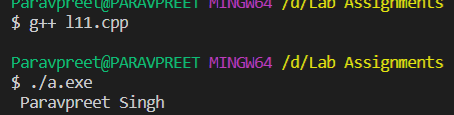
1. ‘\r’ is carriage return to go to the beginning of the current line

Code:



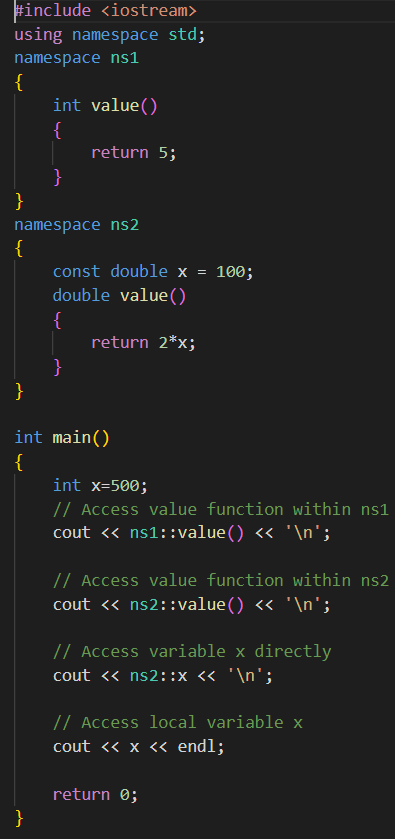
Output:

cout will print Myself and the \r will set the cursor back to the beginning of the line and Paravpreet Singh will overwrite whatever has been printed on that line. And since Paravpreet Singh longer than Myself, all you see is:

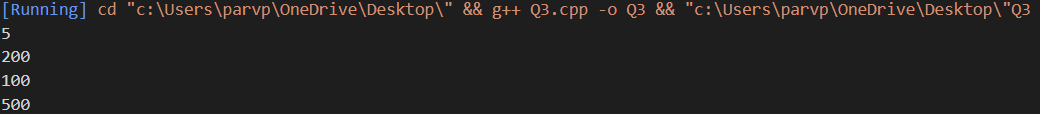


1. Implement *namespace* in a program to illustrate the use of same name variables and functions in different sections/libraries of the code.

Code:



Output:



1. Write a program to define a structure Student that contains 3 variables and 2 functions (a) Name (b) Roll (c) Marks and two functions (1) setStudentData(…) (2) getStudentData(…). Use loops to input and output 3 student object entries.

struct Student{

char name[20]; //string is also okay

int roll; float marks;

void setStudentData(){…}

void getStudentData(){…}

int main(){

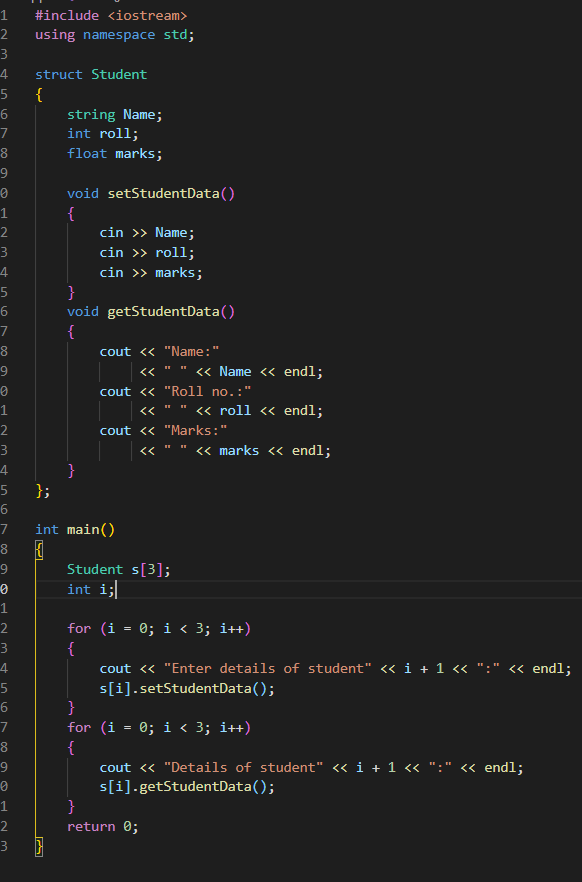
Student s[3];

for(…){ set student data}

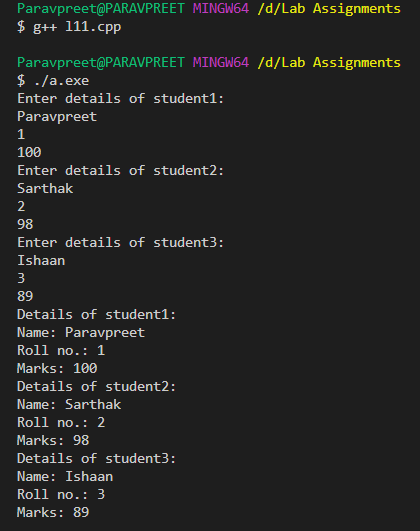
for(…) {get student data}

}

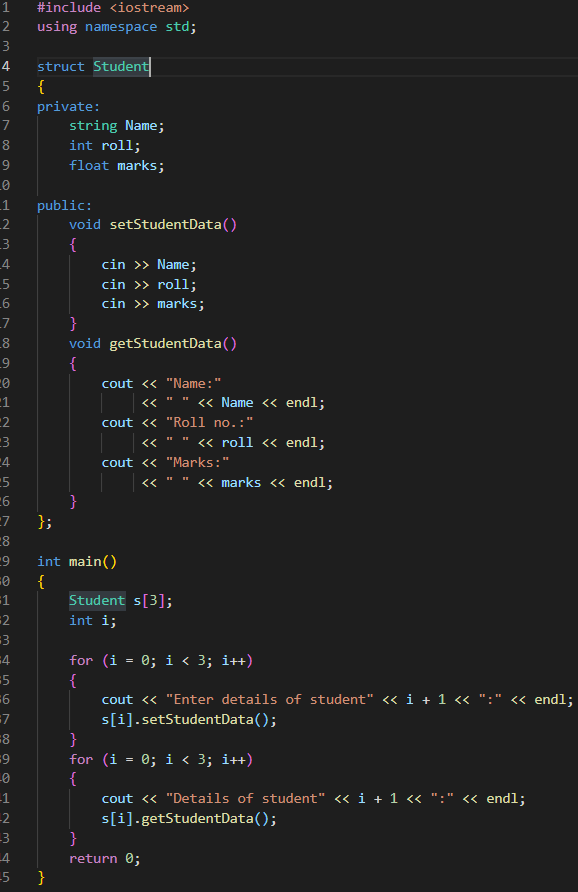
Code :



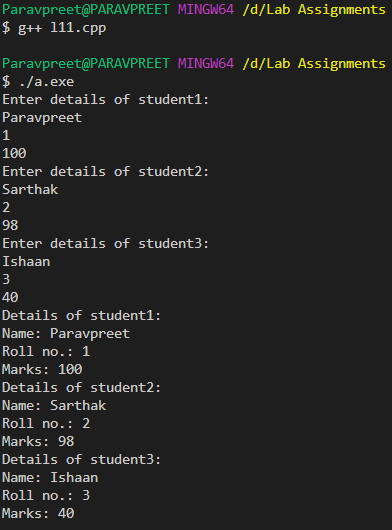
Output:



1. In the previous program make the data private, and functions public.



Output:

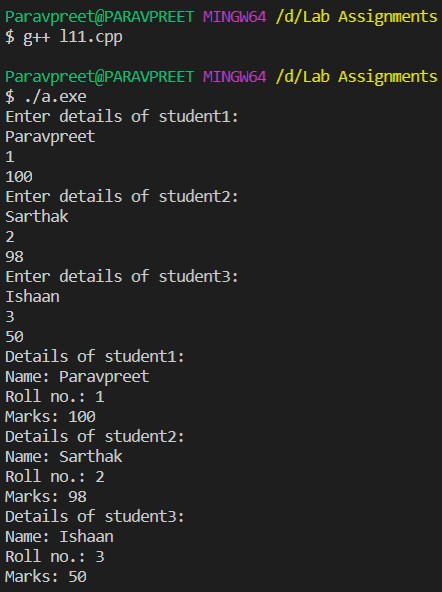


1. Convert program #4 from struct to class.

Code:



Output:



1. Discuss the differences of structures in C and structures in C++ during the lab with other students or your instructor.

Ans:

|  |  |
| --- | --- |
| **Structures** | **Classes** |
| If access specifier is not declared explicitly, then by default access specifier will be public. | If access specifier is not declared explicitly, then by default access specifier will be private. |
| Syntax of Structure:  struct structure\_name { // body of the structure. } | Syntax of Class:  class class\_name { // body of the class. } |
| The instance of the structure is known as "Structure variable". | The instance of the class is known as "Object of the class". |