EUROPEAN UNIVERSITY OF LEFKE

Faculty of Engineering

Department of Computer Engineering



COMP218

OBJECT-ORIENTED PROGRAMMING

**LAB WORK NO. 1**

Prepared by **David O. Ladipo** (174574)

Submitted to Dr. Ferhun Yorgancıoğlu

**Task - 1(a)**

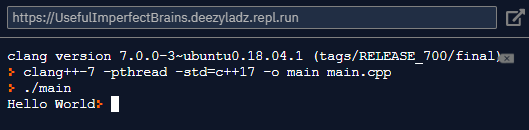
#include <iostream>

int main()

{

std::cout << "Hello World"; // prints Hello world

    return 0;

}

**Task - 1(b)**

#include <iostream>

using std::cout;

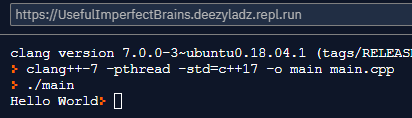
int main()

{

cout << "Hello World"; // prints Hello world

return 0;

}



**Task - 1(c)**

#include <iostream>

using namespace std;

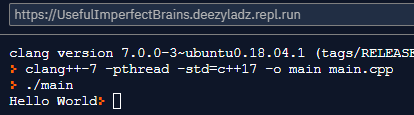
int main()

{

cout << "Hello World"; // prints Hello world

return 0;

}

****

**Task – 2**

#include <iostream>

using namespace std;

int main()

{

int firstNum, secondNum, sumOfTwoNum;

cout << "Enter First Number: " << endl;

cin >> firstNum; //value of first number input from user stored in firstNum

cout << "Enter Second Number: " << endl;

cin>> secondNum; //value of second number input from user stored in secondNum

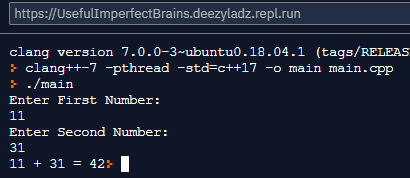
// sum of two numbers in stored in variable sumOfTwoNumbers

sumOfTwoNum = firstNum + secondNum;

// Prints sumOfTwoNum

cout << firstNum << " + " << secondNum << " = " << sumOfTwoNum;

}



**Task – 3**

#include <iostream>

#include <iomanip>

using namespace std;

int main()

{

float x = 3.141559f;

cout << 4 << endl; //prints 4 to the screen

cout << " " << 4 << endl; //prints 2 spaces and 4 to the screen

cout << 4 << " " << endl; //prints 4 and 2 spaces to the screen

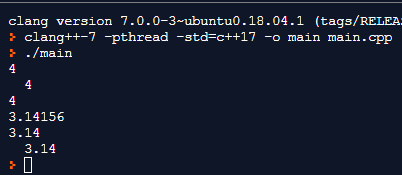
cout << x << endl; //prints float x = 3.141559f;

cout <<fixed<<setprecision(2)<<x<< endl;//rounds up x to 2decimal places

cout <<" "<< fixed << setprecision(2)<< x << endl; //prints to spaces and rounds up x to 2decimal places

return 0;

}

****

**Task – 4**

#include <iostream>

#include <iomanip>

using namespace std;

int main()

{

int a = 2;

char b = 'f';

float c = 3.1415f;

double d = 3;

/\*Set field width (setw):

Sets the field width to be used on output operations.

Behaves as if member width were called with n as argument on the stream on which it is inserted/extracted as a manipulator (it can be inserted/extracted on input streams or output streams).\*/

cout << setw(3) << a << endl; //prints 3 spaces before a

cout << setw(3) << left << a << endl; // prints a then 3 spaces

cout << setw(3) << right << a << endl; // prints 3 spaces before a

cout << '\t' << a << '\t' << b << '\t' << c << endl;

cout << setw(9) << a << setw(8) << b << setw(13) << c << endl;

cout << d << '\t' << setprecision(1) << d << '\t' << fixed << setprecision(1) << d << endl;

cout.unsetf( ios::fixed );

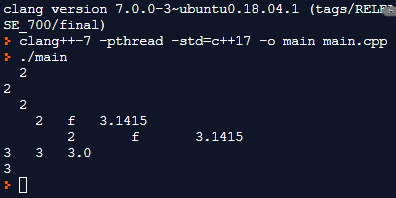
cout << d << endl;

}

/\*Set decimal precision (setprecision)

Sets the decimal precision to be used to format floating-point values on output operations.

Behaves as if member precision were called with n as argument on the stream on which it is inserted/extracted as a manipulator (it can be inserted/extracted on input streams or output streams).\*/



**Task – 5**

#include <iostream>

#include <iomanip>

using namespace std;

/\*Static Cast: This is the simplest type of cast which can be used. It is a compile time cast.It does things like implicit conversions between types (such as int to float, or pointer to void\*), and it can also call explicit conversion functions (or implicit ones).\*/

int main()

{

int a = 3;

char b = 'f';

cout << a << '\t' << static\_cast<char>(a) << endl;

cout << b << '\t' << static\_cast<int>(b) << endl;

cout << ( 2/3 ) << '\t' << ( static\_cast<float>(2) / 3 ) << endl;

}

