**EVEREST ENGINEERING COLLEGE**

  **SANEPA, LALITPUR**

(AFFILIATED TO POKHARA UNIVERSITY)

AN

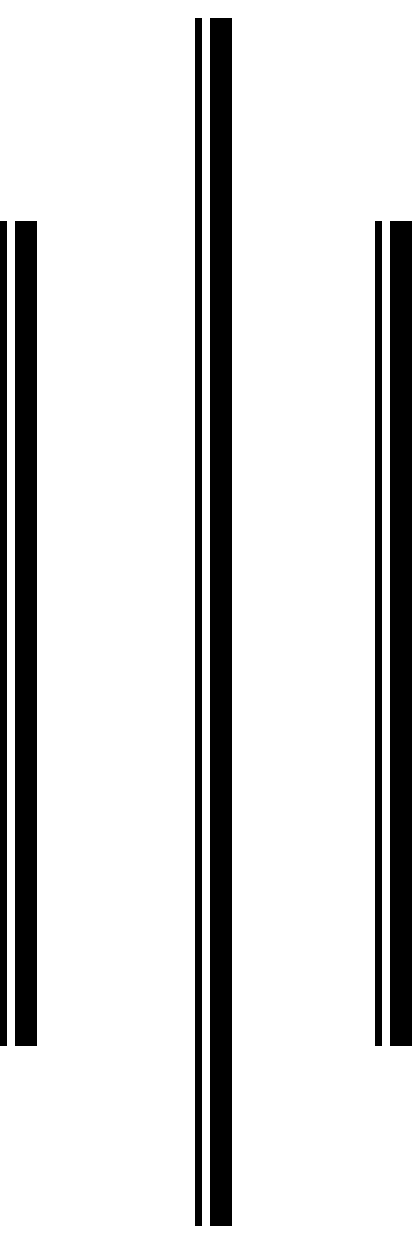
LAB REPORT

ON

**Object Oriented Programming In C++**

***[Basic Introduction In C++]***

**Lab Sheet: 1**



**SUBMITTED BY SUBMITTED TO**

Name:-**SAKCHYAM ACHARYA** Er.Pradip Paudel

Roll No:-40 Everest Engineering College

Department of Computer Engineering

Faculty:-BE CMP

Year/Semester: - 2nd SEM

Batch: 2021

Verified By:-

Student's Signature

**Index Page:-**

Topics Page No

* Title, Objective....................................................................................2
* Theory……………………………………………………………………………….2-3
* Pre-Processor Directives
* Header File
* Namespace
* cin & cout
* Manipulator(endl)
* Const Keywords
* Variables Declaration
* Rules of naming variable
* Questions with Source Code & Output…………………………………………...3-9
* Discussion & Conclusion ...................................................................................9

**Title:-**Basic Introduction to C++.

**Objective:-**To be familiar with the syntax and implement a well-structured, robust computer program using the C++ programming language & to understand its compilation process.

**Theory:-**

**Pre-Processor Directives:-** The preprocessors are the directives, which give instructions to the compiler to preprocess the information before actual compilation starts.

All preprocessor directives begin with #, and only white-space characters may appear before a preprocessor directive on a line. Preprocessor directives are not C++ statements, so they do not end in a semicolon (;).

You already have seen #**include** directive in all the examples. This macro is used to include a header file into the source file.

There are number of preprocessor directives supported by C++ like #include, #define, #if, #else, #line, etc.

**Header File(<iostream>):-** These are those files that store predefined functions. It contains definitions of functions that you can include or import using preprocessor directive #include. This preprocessor directive tells the compiler that the header file needs to be processed prior to the compilation.

**Namespace:-** Namespace in C++ is the declarative part where the scope of identifiers like functions, the name of types, classes, variables, etc., are declared.

**cin & cout:-** cin is an object of the input stream and is used to take input from input streams like files, console, etc. cout is an object of the output stream that is used to show output. Basically, cin is an input statement while cout is an output statement.

**Manipulator (endl):-** The word 'endl' in C++, a programming language, stands for end of line. Furthermore, the use of the endl C++ manipulators takes place to move the cursor to the beginning of the next line. Moreover, its working is similar to the '\n' escape sequence.

**Const Keywords:-** Constant member functions are those functions that are denied permission to change the values of the data members of their class. To make a member function constant, the keyword “const” is appended to the function prototype and also to the function definition header.

**Variables Declaration:-** Variable declaration in C++ is a part which is done in the starting only to ensure the compiler that there is some variable with the given type and name used in the program so that it can proceed with the further compilation without giving any issues. A variable in C++ is declared before its first use in the program.

**Rules of naming variable:-**

**The general rules for naming variables are:**

* Names can contain letters, digits and underscores.
* Names must begin with a letter or an underscore (\_)
* Names are case sensitive ( myVar and myvar are different variables)
* Names cannot contain whitespaces or special characters like !, #, %, etc.
* **Problems with Source Code & Output:-**

**Problem No:1**

**Wap to find the area of circle by taking radius input from user.**

*Source Code:*

//Wap to find the area of circle.

#include<iostream>

#define pi 3.14

using namespace std;

int main(){

float rad;

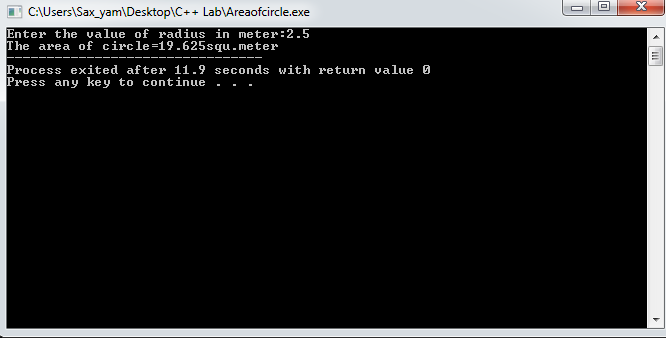
cout<<"Enter the value of radius in meter:";

cin>>rad;

cout<<"The area of circle="<<(pi\*rad\*rad)<<"squ.meter";

return 0;

}



**Problem No:-2**

**Wap to find Simple Interest.**

*Source Code:-*

 //Simple Interest

#include<iostream>

using namespace std;

int main(){

float prin,time,rate,simint;

cout<<"Enter the value of principle in RS:";

cin>>prin;

cout<<"Enter the value of Time in year:";

cin>>time;

cout<<"Enter the value of rate of interest in RS:";

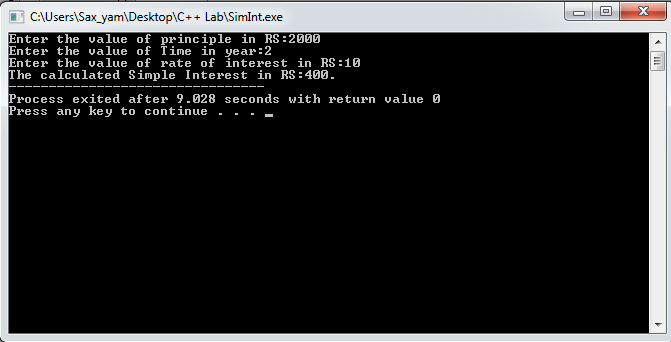
cin>>rate;

simint=(prin\*time\*rate)/100;

cout<<"The calculated Simple Interest in RS:"<<simint<<".";

return 0;

}

****

**Problem No 3:-**

**Wap to find maximum number among three numbers.**

Source Code:

//Wap to find the maximum among three numbers.

#include<iostream>

using namespace std;

int main(){

int n1,n2,n3;

cout<<"Enter 1st,2nd & 3rd number:";

cin>>n1>>n2>>n3;

if((n3>n2) && (n3>n1)){

cout<<endl<<"Greatest Number="<<n3;

}

else if((n1>n3) && (n1>n2)){

cout<<endl<<"Greatest Number="<<n1;

}

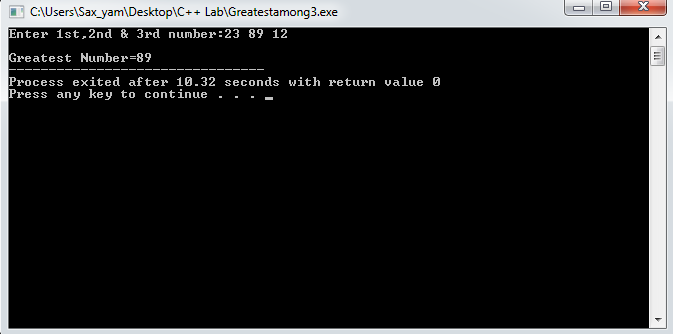
else{

cout<<endl<<"Greatest Number="<<n2;

}

return 0;

}

****

**Problem No:4**

**WAP to find the total percentage obtained by student by taking input of marks obtained in 5 different subject by considering 100 as a full marks.**

*Source Code:-*

//marks of 5 students & find percentage

#include<iostream>

using namespace std;

int main(){

float cpp,thermal,applied,drawing,physics,Totalper;

cout<<" Marks of Sujan In 5 Different Subject";

cout<<endl<<"Marks in C++:";

cin>>cpp;

cout<<endl<<"Marks in Thermal:";

cin>>thermal;

cout<<endl<<"Marks in Applied:";

cin>>applied;

cout<<endl<<"Marks in Drawing:";

cin>>drawing;

cout<<endl<<"Marks in Physics:";

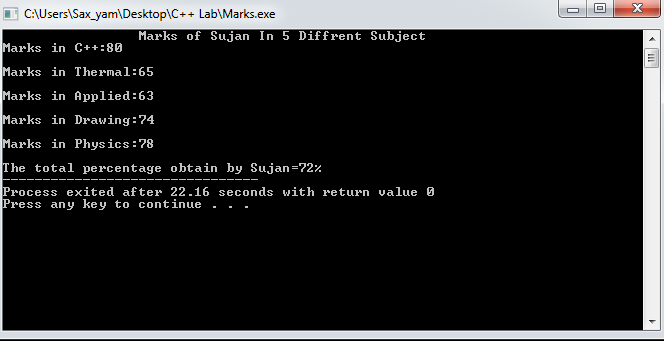
cin>>physics;

Totalper=(((cpp + thermal + applied + drawing + physics)/500)\*100);

cout<<endl<<"The total percentage obtain by Sujan="<<Totalper<<"%";

return 0;

}

****

**Problem No:-5**

**WAP to find the sum of two different numbers by implementing four different categories of function.**

*Source Code:*

/\*Function Type

1.no arguments & no return value

2.no arguments but return value

3.with arguments but no return value

4.with arguments and return value

\*/

#include<iostream>

using namespace std;

void sum1(void){

int n1,n2;

cout<<"Enter 1st & 2nd number:";

cin>>n1>>n2;

cout<<endl<<"Function with no arg & no ret valu="<<(n1+n2)<<endl;

};

int sum2(void){

int n1,n2;

cout<<endl<<"Enter two numbers:";

cin>>n1>>n2;

return(n1+n2);

};

void sum3(int n1,int n2){

cout<<endl<<" Function with argu but no retu valu="<<(n1+n2);

};

int sum4(int n1,int n2){

return(n1+n2);

};

int main(void){

int n1,n2;

cout<<"Enter 1st & 2nd number:";

cin>>n1>>n2;

cout<<endl<<"Function without arg but return value="<<sum2();

//cout<<"Function no arguments & no return value="<<sum1(n1,n2)<<endl;

//cout<<"Function no arguments but return value="<<sum2(n1,n2)<<endl;

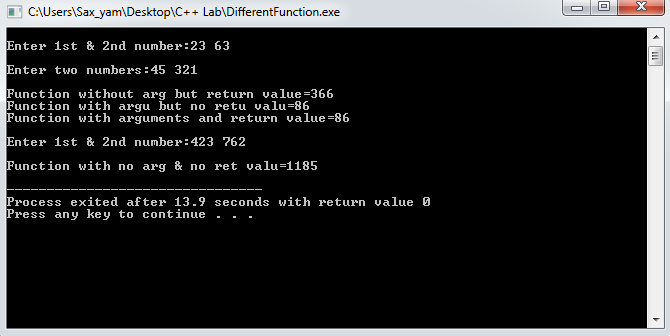
sum3(n1,n2);

cout<<endl<<"Function with arguments and return value="<<sum4(n1,n2)<<endl;

sum1();

return 0;

}

****

**Problem No:-6**

**WAP to find the Simple Interest by implementing the four different categories of functions.**

*Source Code:-*

//Wap to find SI by implementing all categories of function

/\*Function Type

1.no arguments & no return value

2.no arguments but return value

3.with arguments but no return value

4.with arguments and return value

\*/

#include<iostream>

using namespace std;

void SI1(void){

int p,t,r;

cout<<"Enter Prinicple in RS,Time in Year,Rate of interest in %:";

cin>>p>>t>>r;

cout<<endl<<"SI from no arguments & no return value="<<((p\*t\*r)/100)<<endl;

};

int SI2(void){

int p,t,r;

cout<<"Enter Prinicple in RS,Time in Year,Rate of interest in %:";

cin>>p>>t>>r;

return((p\*t\*r)/100);

};

void SI3(int p,int t,int r){

cout<<endl<<" SI with argu but no retu valu="<<((p\*t\*r)/100);

};

int SI4(int p,int t,int r){

return((p\*t\*r)/100);

};

int main(){

int p,t,r;

cout<<"Enter Prinicple in RS,Time in Year,Rate of interest in %:";

cin>>p>>t>>r;

cout<<endl<<"SI without arg but return value="<<SI2();

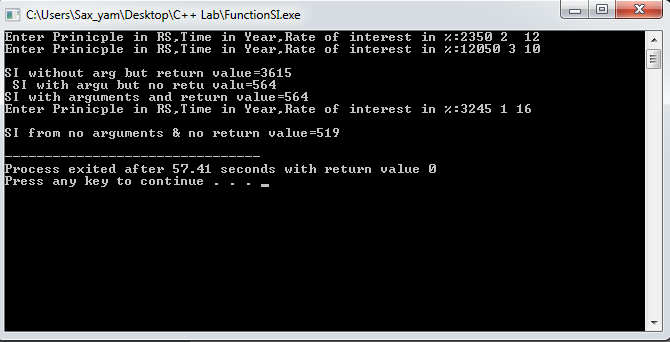
SI3(p,t,r);

cout<<endl<<"SI with arguments and return value="<<SI4(p,t,r)<<endl;

SI1();

return 0;

}



**Discussion & Conclusion:-**

The program is focused on calculation of Area of the circle of given radius, Simple interest, greatest number among three numbers, sum of two different numbers by implementing 4 different categories of function.

From this program I understood the difference of int & float, how to deal with C++ program, different operators in c++, conditional statements in c++, & how to use different categories of function.

**Thank You**<SAKWheels>