1.Declare a class student with

Data member:

- Roll no

- Name

- Marks in five subject

- Total

- Percentage

Member function:

- Input() to input data

- Display() to display data

- Calavg() to calculate average Enter data for 5 students.

#include<iostream>

#include<iomanip>

using namespace std;

class student

{

private:

int roll\_no;

float marks[10],total=0,percent;

char name[20];

public:

void input();

void display();

void calAvg();

};

void student :: input()

{

cout<<"Enter roll no. :";

cin>>roll\_no;

cout<<"Enter name :";

cin>>name;

cout<<"Enter marks of 5 subjects :";

for(int i=0;i<5;i++)

{

cin>>marks[i];

total+=marks[i];

}

percent=total/5;

}

void student :: display()

{

cout<<"Roll no:"<<setw(8)<<roll\_no<<endl;

cout<<"Name:"<<setw(8)<<name<<endl;

cout<<"Marks of 5 subjects are :"<<endl;

for(int i=0;i<5;i++)

{

cout<<marks[i]<<" ";

}

cout<<"\nTotal ="<<setw(8)<<total<<endl;

cout<<"Percentage ="<<setw(8)<<percent<<"%"<<endl;

}

void student :: calAvg(){

float avg;

avg=total/5;

cout<<"Average ="<<setw(8)<<avg;

}

int main()

{

student s1[5],s2;

for(int i=0;i<2;i++)

{

cout<<"\nEnter data of student"<<i+1<<endl;

s1[i].input();

}

cout<<endl;

for(int i=0;i<2;i++)

{

cout<<"\nData of student"<<i+1<<endl;

s1[i].display();

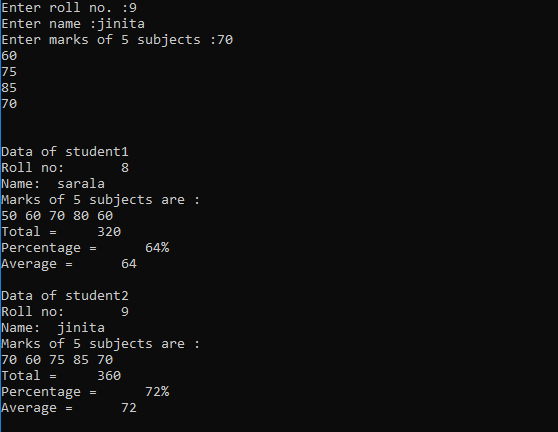
s1[i].calAvg();

cout<<endl;

}

return 0;

}



1. Write a menu driven program to calculate area of rectangle, triangle, and circle using concept of function overloading.

#include<iostream>

#include<iomanip>

#define PI 3.14

using namespace std;

void area(int l,int b)

{

int area=l\*b;

cout<<"Area of rectangle ="<<setw(5)<<area<<endl<<endl;

}

void area(float b,float h)

{

float area,s;

area=0.5\*b\*h;

cout<<"Area of triangle ="<<setw(5)<<area<<endl<<endl;

}

void area(float r)

{

float area;

area= PI\*r\*r;

cout<<"Area of circle ="<<setw(5)<<area<<endl<<endl;

}

int main()

{

int len=15,br=14;

int choice=1;

float base=9,height=8,radius=7;

while(choice!=0)

{

cout<<"Enter choice (1-4):"<<endl;

cin>>choice;

switch(choice)

{

case 1:

area(len,br);

break;

case 2:

area(base,height);

break;

case 3:

area(radius);

break;

case 4:

exit(0);

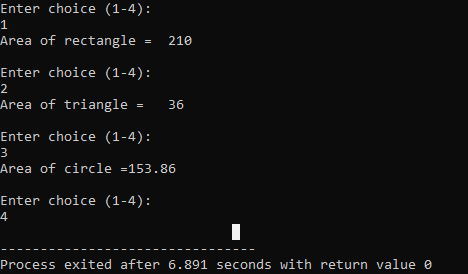
default:

cout<<"Invalid Choice!!"<<endl<<endl;

}

}

return 0;

}

. 3. Write a program to input the salary of person and calculate the HRA and DA according to the following conditions.

Salary HRA DA

<5000 5% 2%

5000-10000 10% 5%

10001-15000 15% 8%

>15000 17% 10%

#include<iostream>

#include<iomanip>

using namespace std;

int main()

{

float salary,hdr,da;

cout<<"Enter salary:"<<endl;

cin>>salary;

if(salary<5000){

hdr=0.05\*salary;

cout<<"HDR ="<<setw(4)<<hdr<<endl;

da=0.02\*salary;

cout<<"DA ="<<setw(4)<<da<<endl;

salary+=hdr+da;

cout<<"Total salary="<<setw(4)<<salary<<endl;

}

else if(salary>=5000&&salary<10000)

{

hdr=0.1\*salary;

cout<<"HDR ="<<setw(4)<<hdr<<endl;

da=0.05\*salary;

cout<<"DA ="<<setw(4)<<da<<endl;

salary+=hdr+da;

cout<<"Total salary="<<setw(4)<<salary<<endl;

}

else if(salary>=10000&&salary<15000)

{

hdr=0.15\*salary;

cout<<"HDR ="<<setw(4)<<hdr<<endl;

da=0.08\*salary;

cout<<"DA ="<<setw(4)<<da<<endl;

salary+=hdr+da;

cout<<"Total salary="<<setw(4)<<salary<<endl;

}

else

{

hdr=0.17\*salary;

cout<<"HDR ="<<setw(4)<<hdr<<endl;

da=0.1\*salary;

cout<<"DA ="<<setw(4)<<da<<endl;

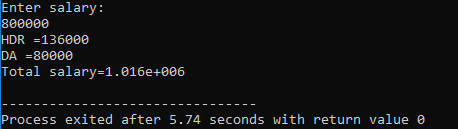
salary+=hdr+da;

cout<<"Total salary="<<setw(4)<<salary<<endl;

}

return 0;

}





**LAB ASSIGNMENT 4**

SUBMITTED BY: SUBMITTED TO: sarala phuyal Jeeban Dhungel

Roll No.: 19

1. Write a program to create a class account with data members account no, account

holder’s name and balance. Add member functions create account, deposit, withdraw and balance inquiry to the class. Create an object of the account class and show all operation with this.

#include<iostream>

#include<iomanip>

using namespace std;

class bank

{

private:

char name[20];

int acc\_no;

char acc\_type[10];

float balance;

public:

void create\_account();

void deposit();

void withdraw();

void balance\_enquiry();

};

void bank :: create\_account()

{

cout<<"Enter account name :"<<endl;

cin>>name;

cout<<"Enter account number:"<<endl;

cin>>acc\_no;

cout<<"Enter account type :"<<endl;

cin>>acc\_type;

cout<<"Enter balance:"<<endl;

cin>>balance;

}

void bank :: deposit()

{

int dep;

cout<<"Enter deposit amount:"<<endl;

cin>>dep;

balance=balance+dep;

}

void bank :: withdraw()

{

int wd;

cout<<"Enter withdraw amount:"<<endl;

cin>>wd;

balance=balance-wd;

}

void bank :: balance\_enquiry()

{

cout<<"Account name:"<<setw(8)<<name<<endl;

cout<<"Account number:"<<setw(8)<<acc\_no<<endl;;

cout<<"Account type :"<<setw(8)<<acc\_type<<endl;

cout<<"Balance :"<<setw(8)<<balance<<endl;

}

int main()

{

bank b1;

int choice=1;

while(choice!=0)

{

cout<<"Enter choice (1-5):"<<endl;

cin>>choice;

switch(choice)

{

case 1:

b1.create\_account();

break;

case 2:

b1.deposit();

break;

case 3:

b1.withdraw();

break;

case 4:

b1.balance\_enquiry();

break;

case 5:

exit(0);

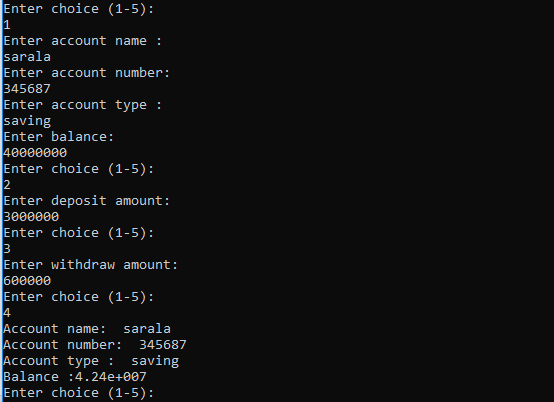
default:

cout<<"Illegal option"<<endl;

}

}

}



2. Define a class matrix with

Data member:

- Two dimensional array for 3\*3 matrix

Member functions:

- To read matrix

- To display matrix in matrix format

- To multiply two matrices

#include<iostream>

using namespace std;

#include<conio.h>

class matrix

{

int a[10][10];

int m,n;

public:

void input();

void output();

void multiply(matrix,matrix);

};

void matrix::input()

{

cout<<"Enter the number of row and columns: ";

cin>>m>>n;

cout<<"Matrix"<<"\n";

for(int i=0;i<m;i++) {

for(int j=0;j<n;j++) {

cin>>a[i][j];

}

}

}

void matrix :: output() {

for(int i=0;i<m;i++) {

cout<<"\n";

for(int j=0;j<n;j++) {

cout<<a[i][j]<<"\t";

}

}

}

void matrix :: multiply(matrix m1, matrix m2)

{

if(m1.n!=m2.m){

cout<<"matrix multiplication is not possible";

}

else

{

for(int i=0;i<m1.m;i++){

for(int j=0;j<m2.n;j++) {

a[i][j]=0;

for(int k=0;k<m1.n;k++) {

a[i][j]=a[i][j] +( m1.a[i][k]\*m2.a[k][j]);

m=m1.m;

n=m2.n;

}

}

}

}

}

int main()

{

matrix m1,m2,m3;

m1.input();

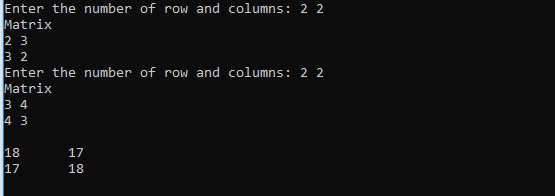
m2.input();

m3.multiply(m1,m2);

m3.output();

getch();

}



3. Define a class distance with

Data members:

- Feet in inch

- Inches in float

Member function:

- To read data member

Friend function:

- To display distance in feet’-inches” format

- To add two distance object and returning distance object

#include<iostream>

#include<iomanip>

using namespace std;

class dist

{

private:

int feet,inches;

public:

void readData();

friend dist display(dist);

friend dist add (dist,dist);

};

void dist :: readData()

{

cout<<"Enter the distance in feet and inches"<<endl;

cin>>feet>>inches;

}

dist display(dist c)

{

cout<<c.feet<<"feet"<<setw(8)<<c.inches<<"inches"<<endl<<endl;;

return c;

}

dist add (dist x, dist y)

{

dist t;

t.inches=x.inches+y.inches;

t.feet=t.inches/12;

t.inches=t.inches%12;

t.feet=x.feet+t.feet+y.feet;

return t;

}

int main()

{

dist d1,d2,d3;

d1.readData();

cout<<"For 1st distance"<<endl;

display(d1);

d2.readData();

cout<<"For 2nd distance"<<endl;

display(d2);

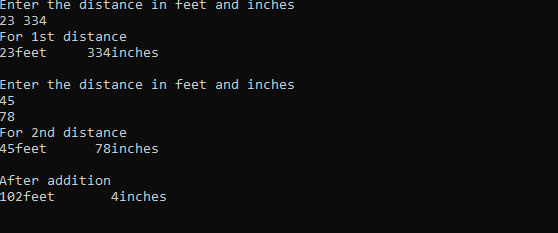
d3=add(d1,d2);

cout<<"After addition"<<endl;

display(d3);

return 0;

}



4.Write a program to check whether a number is palindrome or not.

#include<iostream>

using namespace std;

class palindrome

{

public:

int n,num,rev,rem;

void palindrome\_check();

};

void palindrome :: palindrome\_check()

{

rev=0,num=n;

while(num!=0)

{

rem=num%10;

num=num/10;

rev=rev\*10+rem;

}

if(rev==n)

{

cout<<n<<" "<<"is palindrome." <<endl;

}

else{

cout<<n<<" "<<"is not palindrome." <<endl;

}

}

int main()

{

palindrome a;

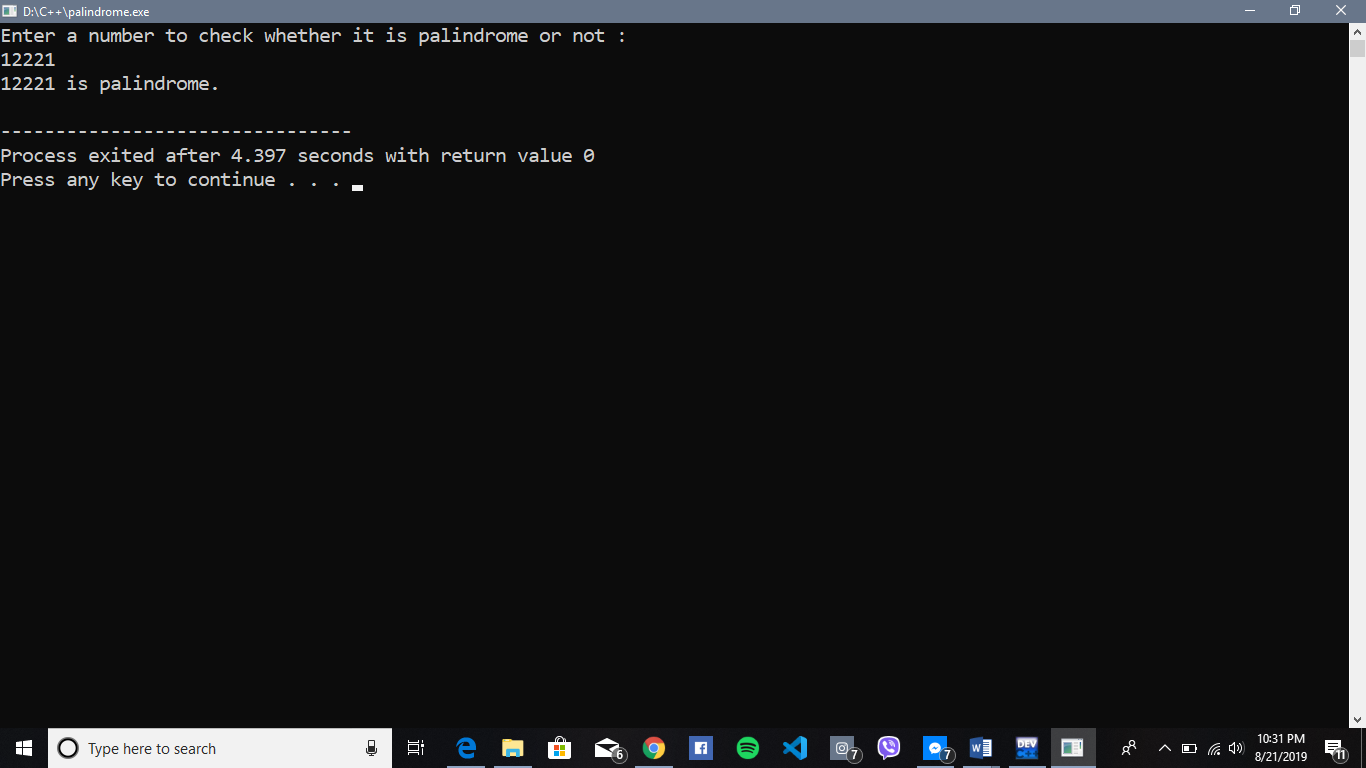
cout<<"Enter a number to check whether it is palindrome or not :"<<endl;

cin>>a.n;

a.palindrome\_check();

return 0;

}



5.Write a program to generate following pattern

\*\*\*\*\*

\*\*\*\*

\*\*\*

\*\*

\*

#include<iostream>

using namespace std;

int main()

{

int i,j,row;

for(i=1;i<=5;i++)

{

for(j=1;j<=5;j++)

{

if(j>=i)

cout<<"\*";

else

cout<<" ";

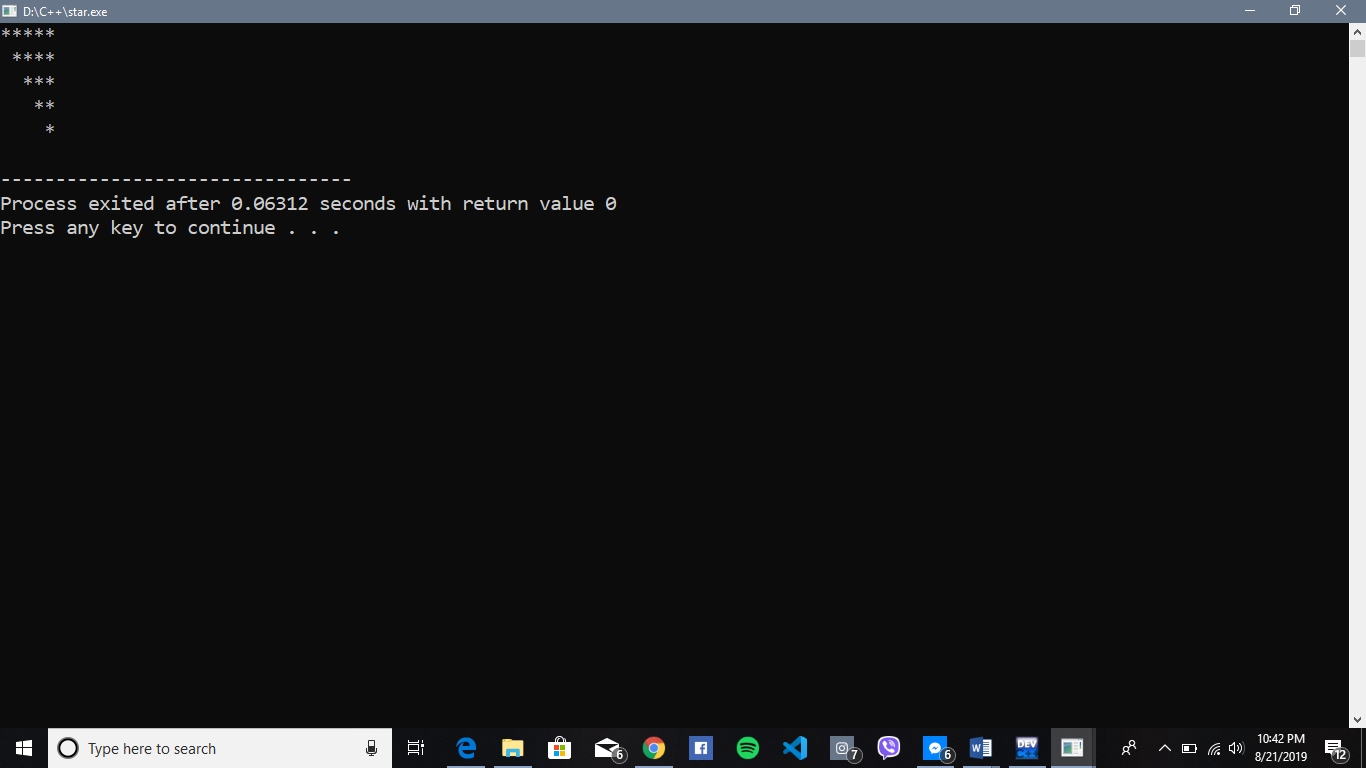
}

cout<<endl;

}

return 0;

}





**LAB ASSIGNMENT 3**

SUBMITTED BY: SUBMITTED TO: sarala phuyal Jeeban Dhungel

Roll No.: 19