Section : B

Lab : 11

Name : Muhammad usama.

Roll no : 17F-8195

Task 1:

#include<iostream>

using namespace std;

int main()

{

char c=0;

int devisor=0;

int devident=0;

try

{

cout<<"eneter devisor :";

cin>>devisor;

cout<<"enetr devident :";

cin>>devident;

if(devisor==0)

{

throw devisor;

}

}

catch(int devisor)

{

cout<<"exception : ";

cout<<"devide by :"<<devisor<<endl;

}

try

{

cout<<"i : for integer throwing : "<<endl;

cout<<"c : for character throwing : "<<endl;

cout<<"s : for string throwing : "<<endl;

cout<<"f : for throwing float : "<<endl;

cin>>c;

if(c=='i')

{

throw 0;

}

if(c=='c')

{

throw c;

}

if(c=='f')

{

throw 0.3;

}

if(c=='s')

{

throw "usoa";

}

}

catch(int i)

{

cout<<"intger exception occure : "<<endl;

}

catch(char c)

{

cout<<"character exception occure : "<<endl;

}

catch(float f)

{

cout<<"float exception occure : "<<endl;

}

catch(string s)

{

cout<<"string exception occure : "<<endl;

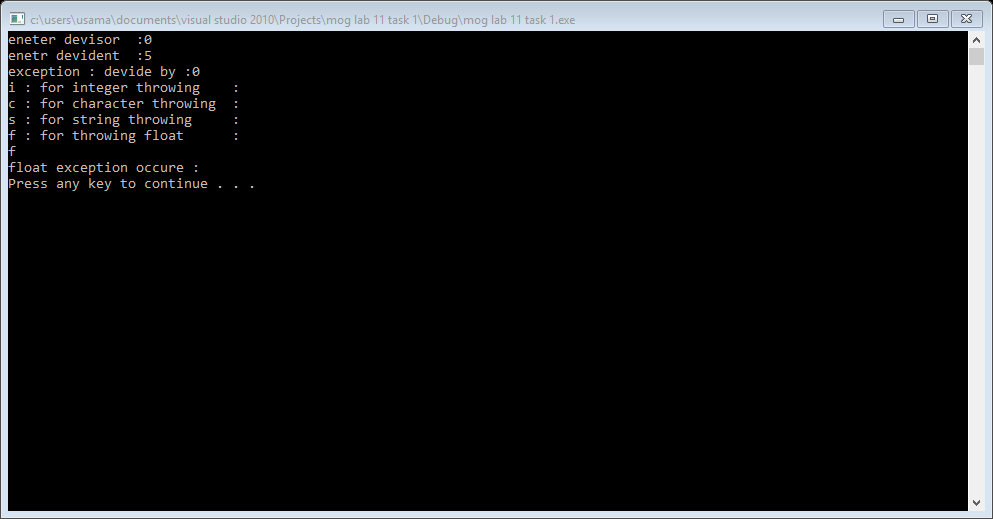
}

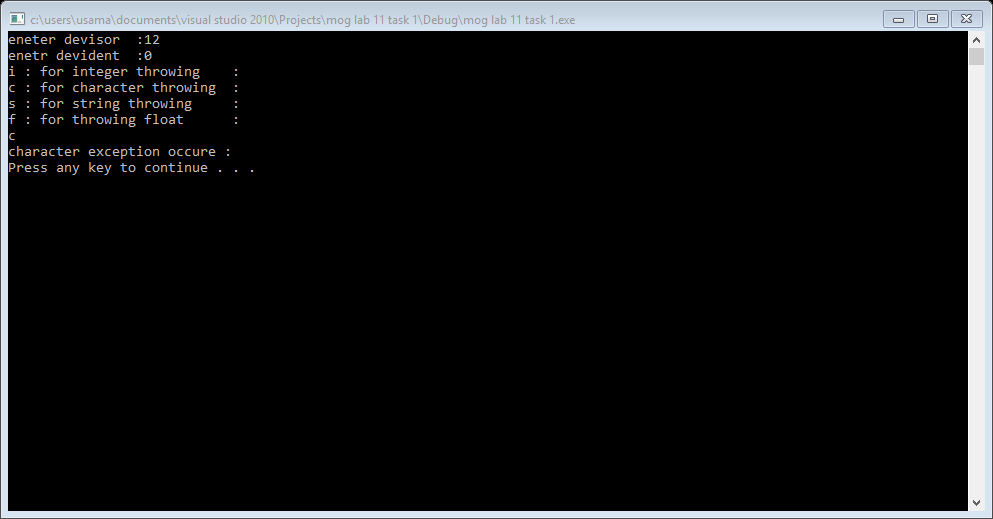
system("pause");

return 0;

}

Result:





Task 2:

#include<iostream>

using namespace std;

int main()

{

float feet=0;

float inches=0;

float centimeter=0;

bool c=true;

do

{

try

{

cout<<"enter feet : ";

cin>>feet;

if(cin.fail())

{

cin.clear();

cin.ignore();

throw "insertiion fail";

}

if(feet<0)

{

throw feet;

}

c=false;

}

catch(...)

{

cout<<"exception occure"<<endl;

cout<<"enetr valid input "<<endl;

}

}while(c);

c==true;

do

{

try

{

cout<<"enter inches : ";

cin>>inches;

if(cin.fail())

{

cin.clear();

cin.ignore();

c=true;

throw "insertiion fail";

}

if(inches<0)

{

c=true;

throw inches;

}

c=false;

}

catch(...)

{

cout<<"exception occure"<<endl;

cout<<"enetr valid input "<<endl;

}

}while(c);

centimeter=(feet\*30.48)+(inches\*2.54);

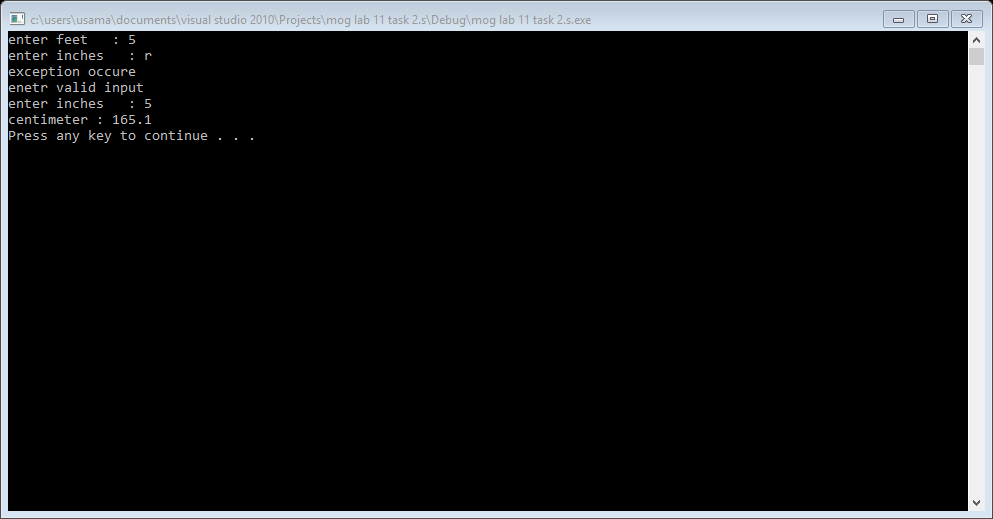
cout<<"centimeter : "<<centimeter<<endl;

system("pause");

return 0;

}

Result:



Task 3:

#include<iostream>

using namespace std;

void addfraction();

void multiplication();

void subtractfraction();

void devision();

void reducefraction(int &x, int &y);

int main()

{

char c;

do

{

cout<<"d : for devision"<<endl;

cout<<"m : for multiplication"<<endl;

cout<<"s : for subtraction"<<endl;

cout<<"a : for addfraction"<<endl;

cout<<"e : for exit"<<endl;

cin>>c;

if(c=='d')

{

devision();

}

else if(c=='m')

{

multiplication();

}

else if(c=='s')

{

subtractfraction();

}

else if(c=='a')

{

addfraction();

}

else if(c=='e')

{

break;

}

else

{

cout<<"invalid input : "<<endl;

}

}while(c!='e');

system("pause");

return 0;

}

void devision()

{

int n1=0,n2=0,n3=0,d1=0,d2=0,d3=0;

cout<<"enetr numerator 1st fraction : ";

cin>>n1;

do

{

cout<<"enetr denomirator 1st fraction : ";

cin>>d1;

try

{

if(d1<=0)

{

throw d1;

}

}

catch(int d)

{

cout<<"eception invalid input"<<endl;

cout<<"you enetr denominator is :"<<d<<endl;

}

}

while(d1<=0);

cout<<"enetr numerator 2nd fraction : ";

cin>>n2;

do

{

cout<<"enetr denomirator 2nd fraction : ";

cin>>d2;

try

{

if(d2<=0)

{

throw d2;

}

}

catch(int d)

{

cout<<"eception invalid input"<<endl;

cout<<"you enetr denominator is :"<<d<<endl;

}

}

while(d2<0);

n3=(n1\*d2);

d3=d1\*n2;

reducefraction(n3,d3);

cout<<n1<<"/"<<d1<<" / "<<n2<<"/"<<d2<<"="<<n3<<"/"<<d3<<endl;

}

void addfraction()

{

int n1=0,n2=0,n3=0,d1=0,d2=0,d3=0;

cout<<"enetr numerator 1st fraction : ";

cin>>n1;

do

{

cout<<"enetr denomirator 1st fraction : ";

cin>>d1;

try

{

if(d1<=0)

{

throw d1;

}

}

catch(int d)

{

cout<<"eception invalid input"<<endl;

cout<<"you enetr denominator is :"<<d<<endl;

}

}

while(d1<=0);

cout<<"enetr numerator 2nd fraction : ";

cin>>n2;

do

{

cout<<"enetr denomirator 2nd fraction : ";

cin>>d2;

try

{

if(d2<=0)

{

throw d2;

}

}

catch(int d)

{

cout<<"eception invalid input"<<endl;

cout<<"you enetr denominator is :"<<d<<endl;

}

}

while(d2<0);

n3=(n1\*d2)+(n2\*d1);

d3=d1\*d2;

reducefraction(n3,d3);

cout<<n1<<"/"<<d1<<"+"<<n2<<"/"<<d2<<"="<<n3<<"/"<<d3<<endl;

}

void subtractfraction()

{

int n1=0,n2=0,n3=0,d1=0,d2=0,d3=0;

cout<<"enetr numerator 1st fraction : ";

cin>>n1;

do

{

cout<<"enetr denomirator 1st fraction : ";

cin>>d1;

try

{

if(d1<=0)

{

throw d1;

}

}

catch(int d)

{

cout<<"eception invalid input"<<endl;

cout<<"you enetr denominator is :"<<d<<endl;

}

}

while(d1<=0);

cout<<"enetr numerator 2nd fraction : ";

cin>>n2;

do

{

cout<<"enetr denomirator 2nd fraction : ";

cin>>d2;

try

{

if(d2<=0)

{

throw d2;

}

}

catch(int d)

{

cout<<"eception invalid input"<<endl;

cout<<"you enetr denominator is :"<<d<<endl;

}

}

while(d2<0);

n3=(n1\*d2)-(n2\*d1);

d3=d1\*d2;

reducefraction(n3,d3);

cout<<n1<<"/"<<d1<<"-"<<n2<<"/"<<d2<<"="<<n3<<"/"<<d3<<endl;

}

void multiplication()

{

int n1=0,n2=0,n3=0,d1=0,d2=0,d3=0;

cout<<"enetr numerator 1st fraction : ";

cin>>n1;

do

{

cout<<"enetr denomirator 1st fraction : ";

cin>>d1;

try

{

if(d1<=0)

{

throw d1;

}

}

catch(int d)

{

cout<<"eception invalid input"<<endl;

cout<<"you enetr denominator is :"<<d<<endl;

}

}

while(d1<=0);

cout<<"enetr numerator 2nd fraction : ";

cin>>n2;

do

{

cout<<"enetr denomirator 2nd fraction : ";

cin>>d2;

try

{

if(d2<=0)

{

throw d2;

}

}

catch(int d)

{

cout<<"eception invalid input"<<endl;

cout<<"you enetr denominator is :"<<d<<endl;

}

}

while(d2<0);

n3=n1\*n2;

d3=d1\*d2;

reducefraction(n3,d3);

cout<<n1<<"/"<<d1<<"\*"<<n2<<"/"<<d2<<"="<<n3<<"/"<<d3<<endl;

}

void reducefraction(int &x, int &y)

{

int large=0;

int gcd=0;

if(x>y)

{

large=x;

}

else

{

large=y;

}

for(int i=large; i>=2; --i)

{

if(x%i==0 && y%i==0)

{

gcd=i;

break;

}

}

if(gcd!=0)

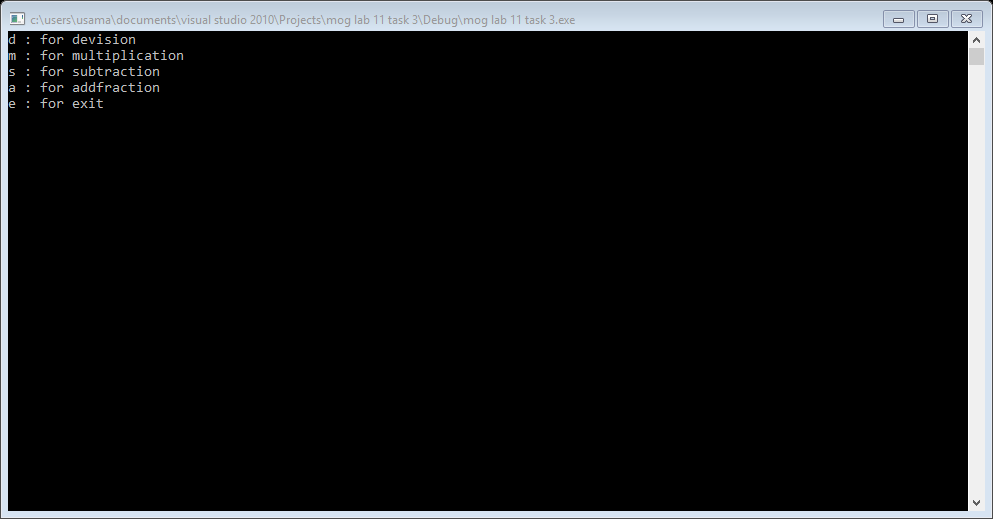
{

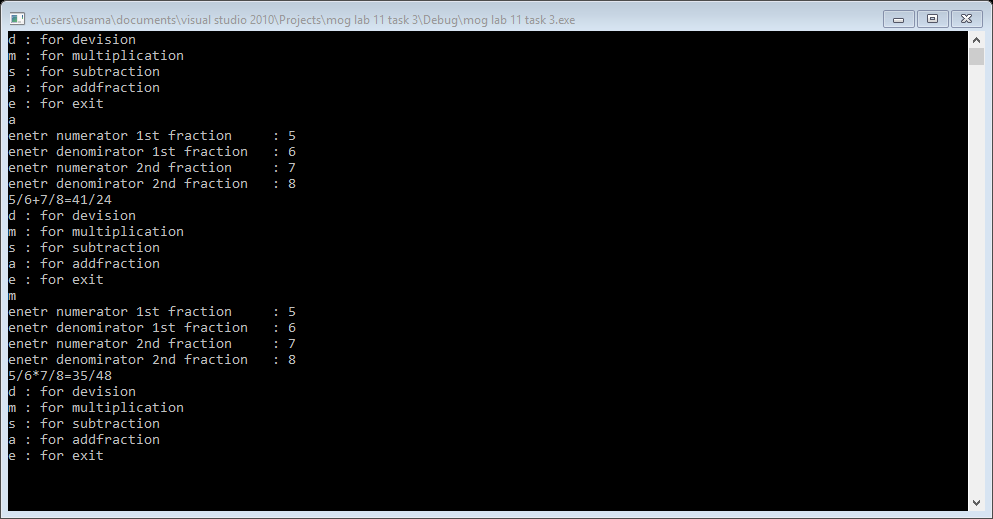
x=x/gcd;

y=y/gcd;

}

}

Result:



Task 4:

#include<iostream>

#include<exception>

using namespace std;

class myexception : public exception

{

int n;

int d;

public:

void setn(int n)

{

this->n=n;

}

void setd(int d)

{

if(d==0)

{

myexception exception;

throw exception;

}

this->d=d;

}

void getdevision()

{

cout<<n<<"/"<<d<<"="<<n/d<<endl;

}

void what()

{

cout<<"the no is devide by zero"<<endl;

}

};

int main()

{

myexception ex;

int n;

int d;

cout<<"enetr numerator : ";

cin>>n;

do

{

cout<<"eneter denomirator : ";

cin>>d;

try

{

ex.setn(n);

ex.setd(d);

}

catch(myexception e)

{

cout<<"exception : ";

e.what();

cout<<"enter valid denomirator:"<<endl;

}

}while(d==0);

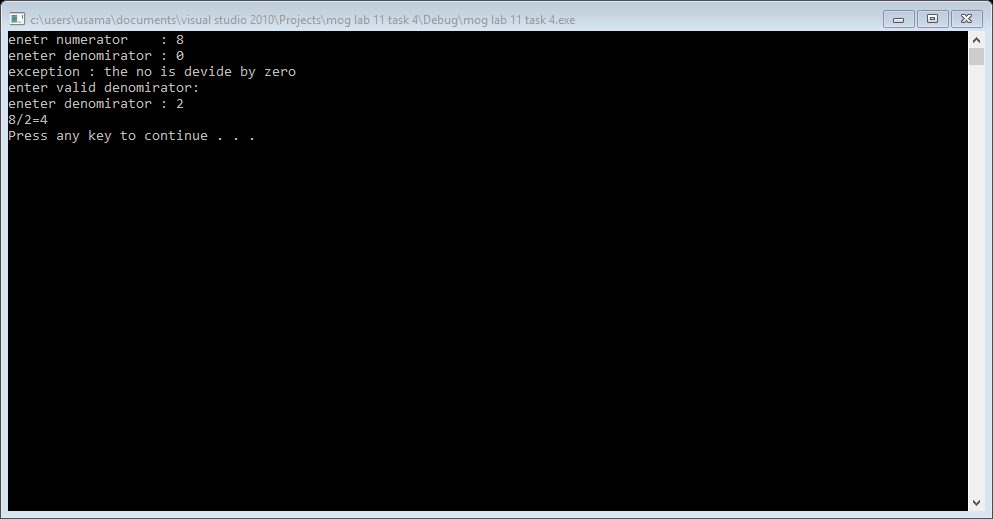
ex.getdevision();

system("pause");

return 0;

}

Result:



Task 5:

#include<iostream>

using namespace std;

template <class t>

void add(t a, t b)

{

cout<<a<<"+"<<b<<"="<<a+b<<endl;

}

template <class t>

void mul(t a, t b)

{

cout<<a<<"\*"<<b<<"="<<a+b<<endl;

}

int main()

{

add(5,10);

add(0.5,0.7);

mul(2,4);

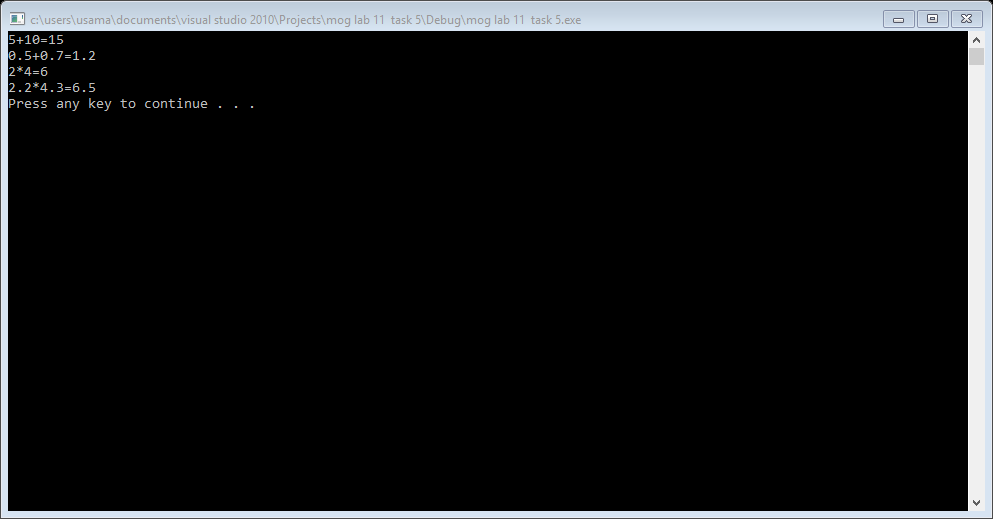
mul(2.2,4.3);

system("pause");

return 0;

}

Result:



Task 6: #include<iostream>

using namespace std;

template<class t>

class area

{

t length;

t width;

t a;

public:

area(t l,t w)

{

a=l\*w;

}

void showarea()

{

cout<<"area : "<<a<<endl;

}

};

int main()

{

area <int>a(2,4);

a.showarea();

area<float>b(2.6,7.4);

b.showarea();

area<float>c(2.6,2);

c.showarea();

area<float>d(2,3.5);

d.showarea();

area<double>e(2,5.9);

e.showarea();

area<double>f(3.9,2);

f.showarea();

area<double>g(3.9,3.3);

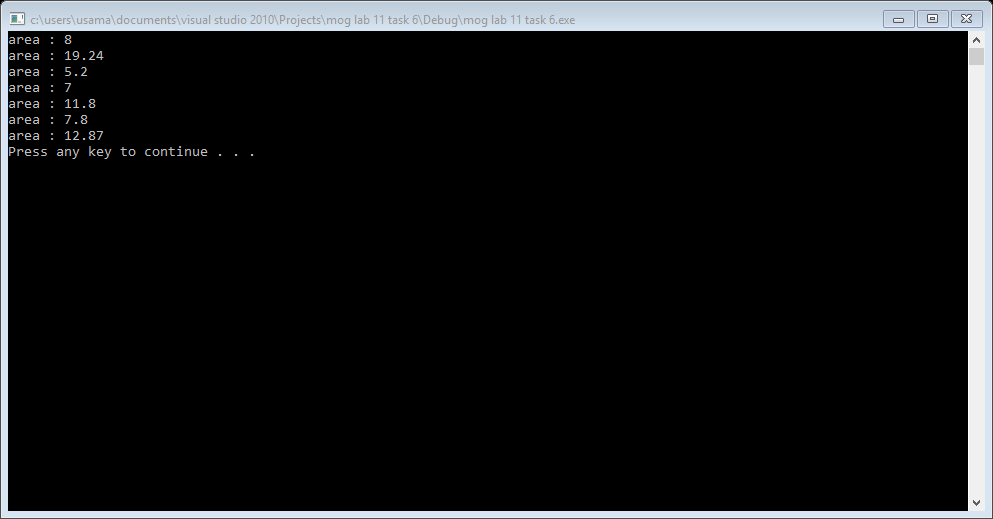
g.showarea();

system("pause");

return 0;

}

Result:



Task 7:

#include<iostream>

#include<math.h>

using namespace std;

template<class t>

class calculator

{

t a;

t b;

public:

calculator(t f ,t s)

{

a=f;

b=s;

}

void addition()

{

cout<<a<<"+"<<b<<"="<<a+b<<endl;

}

void subtraction()

{

cout<<a<<"-"<<b<<"="<<a-b<<endl;

}

void multiplication()

{

cout<<a<<"\*"<<b<<"="<<a\*b<<endl;

}

void devision()

{

cout<<a<<"/"<<b<<"="<<a/b<<endl;

}

void squre\_root()

{

cout<<a<<"=";

cout<<sqrtf(a)<<endl;

}

};

int main()

{

cout<<"addition"<<endl;

calculator<int>a(3,5);

a.addition();

calculator<float>b(8.5,4.6);

b.addition();

calculator<float>c(5.4,4);

c.addition();

calculator<float>d(3,4.4);

c.addition();

cout<<endl<<"subtraction"<<endl;

a.subtraction();

b.subtraction();

c.subtraction();

d.subtraction();

cout<<endl<<"multiplication"<<endl;

a.multiplication();

b.multiplication();

c.multiplication();

d.multiplication();

cout<<endl<<"devision"<<endl;

a.devision();

b.devision();

c.devision();

d.devision();

cout<<endl<<"square root"<<endl;

a.squre\_root();

b.squre\_root();

c.squre\_root();

d.squre\_root();

cout<<endl;

system("pause");

return 0;

}

Result:

