9)Write a program to add 2 ‘Fraction’ objects using operator overloading concept. Implement using a) friend functions b) member functions

//wap to add and subtract two fraction objects (numerator and denominator) using operator overloading concept, using friend functions

#include<iostream>

using namespace std;

class Fraction

{

int num;

int deno;

public:

Fraction()

{

num=0;

deno=1;

}

Fraction(int n,int d)

{

num=n;

deno=d;

}

friend Fraction operator+(Fraction,Fraction);

friend Fraction operator-(Fraction,Fraction);

void display()

{

cout<<num<<"/"<<deno<<endl;

}

};

Fraction operator+(Fraction f1,Fraction f2)

{

Fraction temp;

if(f1.deno!=f2.deno)

{

int f1d=f1.deno;

int f2d=f2.deno;

f1.deno=f1.deno\*f2.deno;

f2.deno=f1.deno\*f2.deno;

f1.num=f1.num\*f2d;

f2.num=f2.num\*f1d;

temp.deno=f1d\*f2d;

}

temp.num=f1.num+f2.num;

return temp;

}

Fraction operator-(Fraction f1,Fraction f2)

{

Fraction temp;

if(f1.deno!=f2.deno)

{

int f1d=f1.deno;

int f2d=f2.deno;

f1.deno=f1.deno\*f2.deno;

f2.deno=f1.deno\*f2.deno;

f1.num=f1.num\*f2d;

f2.num=f2.num\*f1d;

temp.deno=f1d\*f2d;

}

temp.num=f1.num-f2.num;

return temp;

}

int main()

{

Fraction f1(2,2);

Fraction f2(3,2);

Fraction f3;

Fraction f4;

cout<<"The two fractions are:"<<endl;

f1.display();

f2.display();

cout<<"Adding these two fractions and subtracting ....:"<<endl;

f3=f1+f2;

f4=f1-f2;

cout<<"The value of f1 and f2 are:"<<endl;

f1.display();

f2.display();

cout<<"The two fractions f3 and f4 are :"<<endl;

f3.display();

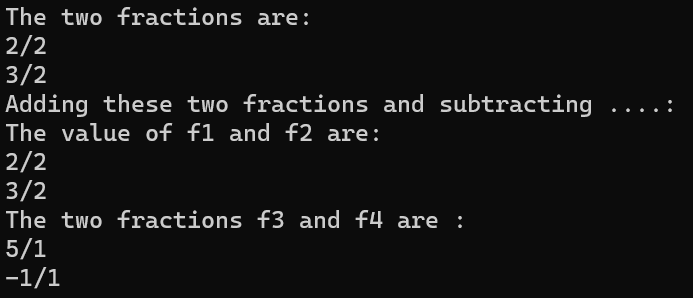
f4.display();

return 0;

}

9)a)

OUTPUT:



b)

//wap to add and subtract two fraction objects (numerator and denominator) using operator overloading concept, use member functions

#include<iostream>

using namespace std;

class Fraction

{

int num;

int deno;

public:

Fraction()

{

num=0;

deno=1;

}

Fraction(int n,int d)

{

num=n;

deno=d;

}

Fraction operator+(Fraction f2)

{

Fraction temp;

int tdeno=deno;

int tnum=num;

temp.deno=deno;

if(tdeno!=f2.deno)

{

tnum=tnum\*f2.deno;

f2.num=f2.num\*tdeno;

temp.deno=tdeno\*f2.deno;

}

temp.num=tnum+f2.num;

return temp;

}

9)b)

Fraction operator-(Fraction f2)

{

Fraction temp;

int tdeno=deno;

int tnum=num;

temp.deno=deno;

if(tdeno!=f2.deno)

{

tnum=tnum\*f2.deno;

f2.num=f2.num\*tdeno;

temp.deno=tdeno\*f2.deno;

}

temp.num=tnum-f2.num;

return temp;

}

void display()

{

cout<<num<<"/"<<deno<<endl;

}

};

int main()

{

Fraction f1(2,2);

Fraction f2(9,3);

Fraction f3;

Fraction f4;

cout<<"The two fractions are:"<<endl;

f1.display();

f2.display();

cout<<"Adding these two fractions and subtracting ....:"<<endl;

f3=f1+f2;

f4=f1-f2;

cout<<"The value of f1 and f2 are:"<<endl;

f1.display();

f2.display();

cout<<"The two fractions f3 and f4 are :"<<endl;

f3.display();

f4.display();

return 0;

}

OUTPUT:

