**LAB ASSIGNMENT**

**WEEK (4-9 May)**

**Q)Define a class ratio with numerator and denominator as member variables.**

**Write the constructors, destructors and copy constructor. Overload the**

**following operators for the same:**

**• Arithmetic operators: +, -, \*, /**

**• Relational operators: <=, >=, ==, !=**

**• Assignment operator: =**

**• Arithmetic assignment operator: +=, -=**

**• Pre and post increment operators: ++**

**• Stream input and output operators: << and >>**

**CODE:-**

#include<iostream>

using namespace std;

class Ratio

{

public:

int num,deno;

Ratio(int n=1,int d=1)

{

num=n;

deno=d;

}

~Ratio(){

}

void print()

{

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

}

Ratio(Ratio &r)

{

num=r.num;

deno=r.deno;

}

Ratio operator\* (Ratio const& obj) {

Ratio res;

res.num = num \* obj.num;

res.deno = deno \* obj.deno;

return res;

}

Ratio operator/ (Ratio & obj) {

Ratio res;

res.num = num \* obj.deno;

res.deno = deno \* obj.num;

return res;

}

Ratio operator + (Ratio const& obj) {

Ratio res;

int x,y;

x = num \* obj.deno;

y = deno \* obj.num;

res.num=x+y;

res.deno=deno\*obj.deno;

return res;

}

Ratio operator- (Ratio & obj) {

Ratio res;

int x,y;

x = num \* obj.deno;

y = deno \* obj.num;

res.num=x-y;

res.deno=deno\*obj.deno;

return res;

}

int operator==(Ratio &obj )

{

int x=num/deno;

int y=obj.num/obj.deno;

if(x==y)

return 1;

else

return 0;

}

int operator<=(Ratio &obj)

{

int x=num/deno;

int y=obj.num/obj.deno;

if(x<=y)

return 1;

else

return 0;

}

int operator>=(Ratio &obj)

{

int x=num/deno;

int y=obj.num/obj.deno;

if(x>=y)

return 1;

else

return 0;

}

int operator!=(Ratio &obj)

{

int x=num/deno;

int y=obj.num/obj.deno;

if(x!=y)

return 1;

else

return 0;

}

int operator=(Ratio const&obj)

{

num=obj.num;

deno=obj.deno;

}

Ratio operator += (Ratio &obj)

{

int x,y;

x = num \* obj.deno;

y = deno \* obj.num;

num=x+y;

deno=deno\*obj.deno;

}

Ratio operator -= (Ratio &obj) {

int x,y;

x = num \* obj.deno;

y = deno \* obj.num;

num=x-y;

deno=deno\*obj.deno;

}

Ratio operator++(){

num=num+deno;

}

Ratio operator ++ (int) {

num=num+deno;

}

friend ostream & operator << (ostream &out, const Ratio &obj);

friend istream & operator >> (istream &in, Ratio &obj);

};

ostream & operator << (ostream &out, const Ratio &obj)

{

out<<obj.num<<"/"<<obj.deno;

return out;

}

istream & operator >> (istream &in, Ratio &obj)

{

cout << "Enter numerator=";

in >> obj.num;

cout << "Enter denominator=";

in >> obj.deno;

return in;

}

int main()

{

Ratio r1,r2,r3;

cout<<"R1:";

r1.print();

cout<<endl<<"R2:";

r2.print();

cout<<endl<<"R3:";

r3.print();

r3=r1+r2;

cout<<endl<<"R3=R1+R2=";

r3.print();

r3=r1-r2;

cout<<endl<<"R3=R1-R2=";

r3.print();

r3=r1\*r2;

cout<<endl<<"R3=R1\*R2=";

r3.print();

r3=r1/r2;

cout<<endl<<"R3=R1/R2=";

r3.print();

if(r1==r2)

cout<<endl<<"R1 equal to R2.";

if(r1<=r2)

cout<<endl<<endl<<"R1 greater than equal to R2.";

if(r1>=r2)

cout<<endl<<endl<<"R1 less than equal to R2.";

r3=r1+r2;

cout<<endl<<endl<<"R3=R1+R2=";

r3.print();

if(r1!=r3)

cout<<endl<<"R1 not equal to R3.";

r3+=r2;

cout<<endl<<endl<<"R3+=R2=";

r3.print();

r3-=r2;

cout<<endl<<"R3-=R2=";

r3.print();

cout<<endl<<"R3:-"<<endl;

cin>>r3;

cout<<endl<<"R3:";

cout<<r3;

r2=r3;

cout<<endl;

r2.print();

return 0;

}