```cpp
#include <iostream>
#include "Rationnel.h"

using namespace std;

int
main()
{
  Rationnel q1, q2(24,-16);

  cout << q1.toString() << " " << q2.toString() << " "
       << Rationnel(231,232).toString()
       << " " << Rationnel(231,-231).toString() << endl<< endl;

  Rationnel t[5];
  for(int i = 0; i<5; i++){
    t[i].setNum(i);
    t[i].setDeno(i+1);
    cout << t[i].toString() << endl;
  }

  cout << endl;

  Rationnel q3(5,6), q4(6,12);
  q3.soustraction(q4, q1);
  cout << q3.toString() << "-" << q4.toString() << "=" << q1.toString()
       << endl;

  return 0;
}
```

```cpp
#ifndef _RATIONNEL_H_
#define _RATIONNEL_H_

#include <string>

using namespace std;

class Rationnel{
private:
  int my_num, my_deno;
public:
  Rationnel();
  Rationnel(int a, int b);
  ~Rationnel();

  void setNum(int a);
  void setDeno(int b);

  string toString() const;
  void simplifie();
  void soustraction(Rationnel autre, Rationnel& difference) const;
};

#endif
```

```cpp
#include <string>
#include "Rationnel.h"
#include "util.h"

using namespace std;

Rationnel::Rationnel()
{
  my_num=0;
  my_deno=1;
}

Rationnel::Rationnel(int a, int b)
{
  my_num=a;
  my_deno=b;
}


Rationnel::~Rationnel()
{
}

void
Rationnel::setNum(int a)
{
  my_num=a;
}

void
Rationnel::setDeno(int b)
{
  my_deno=b;
}


void
Rationnel::simplifie()
{
  int d=pgcd(abs(my_num), abs(my_deno));
  if(d!=0){
    my_num/=d;
    my_deno/=d;
  }
}


string
Rationnel::toString() const
{
  Rationnel q(my_num, my_deno);
  string my_num_s, slash_s, my_deno_s;

  q.simplifie();
  if(q.my_deno<0){
    q.my_num=-q.my_num;
    q.my_deno=-q.my_deno;
  }
  my_num_s=intToString(q.my_num);
  if(q.my_deno==1||q.my_num==0)
    slash_s=my_deno_s="";
  else{
    slash_s="/";
    my_deno_s=intToString(q.my_deno);
  }

  return my_num_s+slash_s+my_deno_s;
```

```cpp
}


void
Rationnel::soustraction(Rationnel autre, Rationnel& difference) const
{
  difference.my_deno=my_deno*autre.my_deno;
  difference.my_num=my_num*autre.my_deno-autre.my_num*my_deno;
}
```

```
#ifndef _UTIL_H_
#define _UTIL_H_

#include <string>
#include <cmath>

using namespace std;

int pgcd(int, int);
string intToString(int);

#endif
```

```
#include <sstream>
#include <string>
#include "util.h"

using namespace std;

int
pgcd(int a, int b)
{
  if(b==0)
    return a;
  else
    return pgcd(b, a%b);
}

string
intToString(int x)
{
  ostringstream oss;
  oss << x;
  string x_s = oss.str();
  return x_s;
}
```