

TD2 – Modularité

1.

```
struct Fraction
{
    int num ;           //numérateur, porte le signe de la fraction
    unsigned int den ; // dénominateur > 0
};
```

2.

Opérateurs binaires : + , - , / , *
Comparaisons : < , > , <= , >= , = , !=
Afficher / Saisir

3.

Squelette du fichier fractions.h

```
#ifndef FRACTIONS_H
#define FRACTIONS_H
```

```
struct Fraction
{
    int num ;           //numérateur, porte le signe de la fraction
    unsigned int den ; // dénominateur > 0
};
```

```
const FRACTIONNULLE = {0,1};
```

```
//Les Constructeurs = sous-programmes permettant de donner une valeur  
initiale à 1 fraction.
```

```
//Les Observateurs = sous-programmes qui donnent 1 information sur 1  
fraction.
```

```
//Les accesseurs = sous-programmes qui donnent accès aux champs/rubriques  
d'une fraction
```

```
//Les opérateurs = additionner () ; soustraire() ;
```

```
//Les opérateurs de comparaisons = estSuperieure() ; estInferieure() ;
```

```
//Autres opérations éventuelles
```

```
//Opérations entrée/sortie
```

```
#endif FRACTIONS_H
```

4.

```
Fraction diviser (Fraction fracG, Fraction fracD) ;  
//BUT : Retourne la fraction irréductible fracG/fracD  
//Pré-cond : fracG et fracD sont irréductibles et fracD != FRACTIONNULLE
```

```
Bool estPositif (Fraction fraction) ;  
//BUT : Retourne vrai si le numérateur est positif sinon retourne faux  
//Pré-cond : fracG et fracD sont irréductibles
```

5.

```
Pour tester estPositif nous allons tester avec :  
Fraction fraction1 = {1,2} ; // Retournera vrai  
Fraction fraction2 = {-2,3} ; //Retournera faux  
//Appel dans le main  
estPositif(fraction1) ; //Retournera True  
estPositif(fraction2) ; //Retournera False  
estPositif(FRACTIONNULLE) ; //Retournera False
```