

TP 3 : classes

Exercice 1 :

Dans cet exercice, on va écrire une classe `rationnel` qui permettra de faire des calculs exacts sur les rationnels.

1. Cette classe devra comprendre :
 - Deux membres privés `num` et `denom`, de type `int` qui correspondront respectivement au numérateur et au dénominateur de la fraction. Les méthodes et fonctions amies de la classe `rationnel` devront toujours renvoyer des résultats sous forme réduite¹.
 - Un constructeur prenant comme argument deux entiers, correspondant au numérateur et au dénominateur, qui auront pour valeurs par défaut respectives 0 et 1.
 - Une méthode publique `valeur`, qui renvoie la valeur numérique (de type `double`) du rationnel.
 - Une surcharge de l'opérateur unaire `-`, qui renverra l'opposé du rationnel considéré.
2. Faut-il écrire le constructeur par copie, l'opérateur `=` et le destructeur ? Si oui, le faire.
3. Surcharger l'opérateur `<<` pour afficher les rationnels sous forme de fraction, ainsi que les opérateurs `+`, `-`, `*` et `/`.
4. Écrire $\sum_{n=1}^{18} \frac{1}{k}$ sous forme de fraction irréductible.
5. La suite $u_0 = 1$, $u_{n+1} = 1 + (1 + u_n)^{-1}$ est une suite de rationnels approchant bien $\sqrt{2}$. Calculez les 10 premiers termes de cette suite.

Exercice 2 :

On va écrire une classe `entier` qui permette de faire des calculs sur des entiers (positifs) arbitrairement grands.

1. Déclarer comme membres privés un tableau `decimales`, contenant des variables de type `int`, et sa taille `nb_chiffres`. Ce tableau devra stocker la suite des décimales.
2. Écrire un constructeur créant un `entier` à partir d'un `unsigned`.
3. Faut-il écrire le constructeur par copie, l'opérateur `=` et le destructeur ? Si oui, le faire.
4. Surcharger les opérateurs `<<`, `+` et `*` pour afficher ces entiers, et faire des opérations dessus.
5. Calculer $200!$.

1. Pour cela, on va devoir écrire une fonction calculant le pgcd de deux entiers.