

Examen

Documents autorisés

On séparera la déclaration de la classe dans le fichier en-tête `fraction.hpp` et le code correspondant dans le fichier `fraction.cpp` pour le code du corps des fonctions.

Attention à la présentation qui devra être parfaitement claire et lisible.

Classe

Définir une classe `Fraction` avec deux entiers `n` et `d` représentant le numérateur et le dénominateur.

Constructeurs et destructeur

Écrire le constructeur vide définissant la fraction à 0/1.

Écrire le destructeur. Expliquer pourquoi il est simple.

Ecrire le constructeur `Fraction::Fraction (int,int)`.

Comment faire pour qu'on puisse définir simplement une fraction avec un entier `Fraction f = 3` (la fraction sera 3/1) sans avoir à écrire `Fraction f = Fraction (3,1)` ?

Opérateurs

Surcharger l'opérateur `-` pour soustraire deux fractions.

Surcharger l'opérateur `*` pour permettre de multiplier une fraction par un entier.

Fonctions membres

Ecrire une procédure membre `Affichage`.

Écrire une fonction membre `Inverse` retournant l'inverse de la fraction. Comment surcharger l'opérateur `/` pour pouvoir écrire `Fraction f = 1 / Fraction(2,3)` ?

On suppose qu'on dispose de la fonction `int Pgcd(int,int)` calculant le plus grand commun diviseur de deux entiers. Ecrire une procédure membre `Simplifie` simplifiant une fraction (par exemple pour 2/6, le plus grand commun diviseur retourné par la fonction est 2 et la fraction doit devenir 1/3).

Fonctions externes

On considère la suite de terme $s_n = 1/1 + \dots + 1/n$. Ecrire la fonction `Terme` calculant le $n^{\text{ème}}$ terme de la série et retournant la fraction simplifiée.