

Le caractère des nombres

février 2009

Ils sont fous ces mathématiciens !

Dans l'antiquité, un certains nombres de mathématiciens (grecs, souvent) jouaient beaucoup avec les nombres entiers. Ils les examinaient, les inspectaient et leur trouvaient toute sorte de propriétés sympathiques.

Par exemple vous apprendrez que 7 est **déficient**, 220 et 284 sont **sociables**, 18 est **abondant** et 496 est **parfait**

1 Les diviseurs (ils ne sont pas nés d'hier)

Tout nombre entier positif (dans \mathbb{N}^*) possède des diviseurs. Au pire il est divisible par 1 et par lui-même (auquel cas il est **premier**). Mais la plupart en ont un peu plus.

Exemple 1: les diviseurs de 8 sont 1, 2, 4 et 8

1. Comment savoir si un nombre p en divise un autre n ?

Aide: `pmodulo`, ça vous rappelle quelque chose ?

2. Complétez la fonction suivante qui stocke dans `d %T` (respectivement `%F`) si p divise (resp. ne divise pas) n :

```
function d=divise(p,n)
// juste une ligne pour vous
// avec un pmodulo quelque part
endfunction
```

3. Écrire un bout de code qui affiche à l'écran tous les diviseurs d'un nombre (42 par exemple)

Aide: pour afficher l'entier n , vous utiliserez `printf("%i ",n)`

4. complétez la fonction suivante qui stocke dans le tableau `diviseurs` tous les diviseurs de n

```
function diviseurs=quels_diviseurs(n)
diviseurs = [ 1 ] ;
// quels sont les autres diviseurs ?
// utilisez la fonction divise
endfunction
```

5. gardez cette fonction bien au chaud, elle est cruciale pour la suite!

2 La somme

Ce qui va nous intéresser dans les exos suivants, c'est la somme des diviseurs d'un nombre. On va donc compléter la fonction suivante

```
function s=somme_diviseurs(n)
// utilisons quels_diviseurs
endfunction
```

En fait on va s'intéresser à la somme des diviseurs **strict** d'un nombre, ce qui exclu ledit nombre.

Exemple 2: les diviseurs de 8 sont 1, 2, 4 et 8 les diviseurs **stricts** de 8 sont 1, 2 et 4

Ajustez le code précédent pour qu'il calcul Écrivez la fonction `somme_diviseurs_stricts` qui ressemble furieusement à la précédente.

1. Combien vaut la somme des diviseurs stricts de 8128 ?
2. Combien vaut la somme des diviseurs stricts de 220 ?
3. Combien vaut la somme des diviseurs stricts de 284 ?

3 Les caractères

Je vais vous donner quelques astuces pour connaître le caractère d'un nombre

- un nombre est **parfait** s'il est égal à la somme de ses diviseurs stricts ;
- un nombre est **déficient** s'il est strictement supérieur à la somme de ses diviseurs stricts ;
- un nombre est **abondant** s'il est strictement inférieur à la somme de ses diviseurs stricts.

1. écrire un bout de code qui teste si un nombre est **parfait**, **déficient** ou **abondant**
2. testez là sur les nombres 8128 220 et 284 (encore eux!)

4 Les amis

Deux nombres sont **amis** si la somme des diviseurs stricts de l'un est égal à l'autre et réciproquement. Testez si 220 et 284 sont amis. Mettre ça dans une fonction qui dit si deux nombres sont amis!