

# Introduction à C/C++

## TD2: Fonctions, Récursivité.

Vincent GAUDILLIERE. [vincent.gaudilliere@inria.fr](mailto:vincent.gaudilliere@inria.fr).

### 1 Hypothénuse.

On considère un triangle rectangle dont les côtés adjacents à l'angle droit ont pour longueurs respectives  $a$  et  $b$ . Ecrivez un programme qui demande à l'utilisateur d'entrer deux valeurs pour  $a$  et  $b$ , puis renvoie la longueur de l'hypothénuse.

(Indication : la fonction `sqrt` de la librairie `<math.h>` permet de calculer la racine carrée d'un nombre. Son prototype est : `double sqrt(double x)`.)

### 2 Nombre premier ?

Ecrire une fonction `is_prime` qui prend en entrée une variable de type `int`, puis renvoie 1 si le nombre est premier, 0 sinon. Le programme doit demander à l'utilisateur d'entrer le nombre à tester, puis appeler la fonction `is_prime`.

### 3 Somme des nombres premiers.

A l'aide de la fonction `is_prime` définie dans l'exercice précédent, calculer (et afficher) la somme de tous les nombres premiers inférieurs à 100000.

### 4 Plus petit multiple.

2520 est le plus petit nombre divisible par tous les nombres allant de 1 jusqu'à 10.

Quel est le plus petit nombre divisible par tous les nombres allant de 1 jusqu'à 20 ?

Pour répondre à cette question, créer une fonction `is_divisible` qui prend en entrée deux entiers  $a$  et  $b$ , et renvoie 1 si  $a$  est divisible par  $b$ , 0 sinon.

## 5 Fonction 91 de McCarthy.

La fonction 91 de McCarthy est une fonction récursive définie pour  $n \in \mathbb{N}$  par :

$$f(n) = \begin{cases} n - 10 & \text{si } n > 100, \\ f(f(n + 11)) & \text{sinon.} \end{cases}$$

Ecrire une fonction McCarthy suivant le modèle décrit ci-dessus. Le programme doit demander à l'utilisateur d'entrer un entier naturel  $n$ , puis afficher la valeur de McCarthy( $n$ ). Que remarquez-vous lorsque  $n \leq 100$  ?

## 6 Puissance.

Soit  $f$  la fonction définie sur  $\mathbb{N} \times \mathbb{N}$  par :

$$f(m, n) = m^n.$$

Implémenter  $f$  sous la forme d'une fonction récursive.

## 7 Toujours plus de récursivité.

Les exercices 8, 9, 10, et 11 du TD1 peuvent tous être implémentés d'une manière plus naturelle à l'aide de fonctions récursives. Le faire.