

# Exercices d'algorithmes

## Algo-1

Un caissier dispose de pièces de 50, 10 et 1 cents. Ecrire un algorithme qui demande un nombre de centimes et indique comment rendre la monnaie sur cette somme, c'est-à-dire combien de chaque pièces le caissier doit rendre. Par exemple, pour 92 cents, il doit afficher "1 pièce de 50 cents" puis "4 pièces de 10 cents" et enfin "2 pièces de 1 cent". Autre exemple, pour 59 cents, il doit afficher "1 pièce de 50 cents" puis "9 pièces de 1 cent".

Il n'est pas demandé de vérifier que le nombre saisi soit positif.

## Algo-2

Ecrire un algorithme qui demande un salaire horaire, puis le nombre d'heures travaillées pendant une semaine. L'algorithme affiche ensuite la paie correspondante sachant que les heures dépassant les 35 heures sont majorées de 30%. Par exemple, si le salaire horaire est de 20€ et que l'utilisateur travaille 40 heures, la paie hebdomadaire est de 830€.

Il n'est pas demandé de vérifier que les nombres saisis soient positifs.

## Algo-3

Simulez l'exécution de l'algorithme suivant et indiquez ce qu'il affiche :

```
Début
  Entier A, B, i

  A ← 5
  B ← 3
  pour i de 3 à 7
    si (A < B)
      A ← A + B % i
    sinon
      B ← B + A/2
    finsi
    afficher A " et " B
  finpour
Fin
```

## Algo-4

On désire afficher la factorielle d'un nombre entier saisi par l'utilisateur (la factorielle de n est le produit de tous les nombres entiers strictement positifs et inférieurs ou égaux à n).

Exemple : la factorielle de 6 vaut 720.

Il n'est pas demandé de vérifier que le nombre saisi soit positif.

1) Quelle structure de contrôle est nécessaire pour faire cela ? Pourquoi ?

2) Ecrire l'algorithme

## Algo-5

L'algorithme suivant doit demander 4 nombres entiers relatifs à l'utilisateur, et afficher pour chacun d'entre eux s'il s'agit d'un multiple de 5 ou non.

Donnez sept raisons pour lesquelles cet algorithme n'est pas bien écrit, ou ne fait pas ce qu'on lui demande (indiquez pour chaque raison le numéro de ligne et le changement à réaliser).

```
1 | Début
2 |   Réel nombre_saisi, reste_division
3 |
4 |   pour i de 0 à 4
5 |     obtenir nombre_saisi
6 |     reste_division = nombre_saisi / 5
7 |     si ( reste_division = 0 )
8 |       afficher nombre_saisi "est un multiple de 5"
9 |     sinon
10 |      afficher nombre_saisi "est un multiple de 5"
11 |   finpour
12 | Fin
```

## Algo-6

La population des Sims Alpha est de 5 000 habitants et elle augmente de 3% par an.

Celle des Sims Beta est de 10 000 habitants et elle augmente de 750 habitants une année sur 2, en commençant par la première (elle ne change pas les autres années).

Ecrire un algorithme permettant de déterminer dans combien d'années la population de Sims Beta dépassera celle des Sims Alpha. Il faut pour cela simuler les populations des deux groupes jusqu'à ce que la condition requise soit atteinte.

On ne se souciera pas des arrondis.

1) Quelle structure de répétition est la plus adaptée ici et pourquoi ?

2) Ecrire cet algorithme

## Algo-7

On veut calculer le pgcd (plus grand commun diviseur) de 2 nombres en utilisant l'algorithme d'Euclide. D'après Wikipedia : cela consiste, pour déterminer le PGCD de deux nombres a et b, à effectuer la division euclidienne de a par b (on trouve alors un reste r1) puis la division euclidienne de b par r1 (on note le reste trouvé r2), puis celle de r1 par r2... Et ainsi de suite. Le PGCD de a et b est le dernier reste non nul de cette suite.

Exemple : pour a=35 et b=20, le pgcd vaut 5

1) Quelle structure de répétition est nécessaire pour faire cela ? Pourquoi ?

2) Ecrire l'algorithme

## Algo-8

Ecrire un algorithme permettant de calculer une moyenne pondérée de 5 notes. Chaque note sera un nombre entier compris entre 0 et 20. L'utilisateur devra saisir à chaque fois une note, puis son coefficient (nombre entier supérieur à 1). L'algorithme s'assurera que les nombres saisis respectent les conditions (en redemandant une saisie si ce n'est pas le cas) et affichera au final la moyenne.