

1) Ne pas perdre la balle des yeux

a) À l'issue de l'algorithme suivant :

```
A ← 1
B ← A + 1
A ← B + 2
B ← A + 2
A ← B + 3
B ← A + 3
```

Quelles sont les valeurs contenues dans les variables A et B ?

b) À l'issue de l'algorithme suivant :

```
N ← 1
S ← N
N ← N + 1
S ← S + N
N ← N + 1
S ← S + N
N ← N + 1
S ← S + N
N ← N + 1
S ← S + N
```

Quelles sont les valeurs contenues dans les variables N et S ?

2) Périmètre et aire d'un rectangle

Écrire un programme demandant à l'utilisateur la longueur et la largeur d'un rectangle, et affichant son périmètre et son aire.

3) Conversion temporelle

Écrire un programme demandant à l'utilisateur un entier s et renvoyant le temps correspondant à s secondes, affiché sous la forme **hh:mm:ss**, **hh** étant le nombre d'heures, **mm** le nombre de minutes et **ss** le nombre de secondes.

4) Rendre la monnaie

On désire écrire un programme aidant à rendre la monnaie. On suppose qu'on dispose d'un nombre illimité de pièces de 1, 2, 5, 10, 20 et 50 centimes. Une somme S (en centimes) étant donnée, on veut la décomposer en le plus petit nombre possible de pièces de monnaie. Par exemple, si l'on doit rendre 93 centimes, le programme devra afficher :

```
Pour rendre 93 centimes, il faut donner :
- 1 pièce de 50 centimes,
- 2 pièces de 20 centimes,
- 0 pièces de 10 centimes,
- 0 pièces de 5 centimes,
- 1 pièces de 2 centimes,
- 1 pièces de 1 centime
```

Vous noterez que l'affichage n'est pas très satisfaisant : il y a des fautes d'orthographe, et des lignes inutiles. Si vous connaissez les conditionnelles, vous pouvez améliorer l'affichage en corrigeant ces défauts.

5) Échange de variables

a) Écrire un algorithme échangeant les contenus de deux variables **a** et **b**.

b) Écrire de même un algorithme échangeant les contenus de trois variables **a**, **b** et **c**, selon le schéma **a -> b -> c -> a**.

c) Le langage Python autorise les affectations "simultanées", de la forme : **a, b = b, a**.

Que fait, selon vous, l'interprète Python lorsqu'il rencontre une telle ligne ? Cette syntaxe vous paraît-elle utile ? efficace ?

d) Que fait l'algorithme suivant :

```
a := a+b;  
b := a-b;  
a := a-b;
```

Quel est l'avantage de cet algorithme par rapport à celui de la première question ? Quel en est le principal inconvénient ?

- e) Concevoir un algorithme similaire répondant à la deuxième question.
- f) Travail personnel : démontrer qu'on ne peut pas échanger les contenus de deux variables en moins de trois instructions.
- g) Par quelle opération pourrait-on remplacer l'addition si les variables à permuter étant de type booléen, ou caractère ?
- h) Travail personnel : concevoir un algorithme effectuant une permutation circulaire sur n variables $a[1], \dots, a[n]$, envoyant $a[1]$ sur $a[k+1]$, $a[2]$ sur $a[k+2]$, etc.

