

# Le pseudo-code

## Exercices

### Exercice 1 :

Écrire un algorithme qui demande à un utilisateur de saisir 3 notes et d'en calculer la moyenne.

L'instruction "SAISIR" demande à l'utilisateur de saisir au clavier une valeur. Exemple : **SAISIR x**

### Exercice 1 bis :

Écrire un algorithme qui demande à un utilisateur de saisir 3 notes et d'en calculer la moyenne.

Vous devez créer une **fonction** qui retourne un nombre entre 0 et 20 saisi au clavier. L'utilisateur doit recommencer sa saisie si le nombre n'est pas compris entre 1 et 20.

L'instruction "SAISIR" demande à l'utilisateur de saisir au clavier une valeur. Exemple : **SAISIR x**

### Exercice 2 :

Écrire un algorithme qui permet de calculer la moyenne de n notes stockées dans une liste

### Exercice 3 :

Écrire un algorithme qui demande à un utilisateur de saisir un nombre en binaire et qui convertit ce nombre binaire en nombre décimal, après sa saisie par un utilisateur.

Exemples :

10 en binaire donne 2 en Décimal

101 en binaire donne 5 en Décimal

11001 en binaire donne 25 en Décimal

### Exercice 4 :

Écrire un algorithme permettant de générer, de manière aléatoire, un nombre entre 1 et 100

L'utilisateur doit ensuite trouver le nombre choisi par l'ordinateur en réalisant des saisies. S'il se trompe, un message d'erreur est affiché pour lui indiquer si le nombre saisi est trop grand ou trop petit.

L'algorithme se termine lorsque le nombre est trouvé.

### Exercice 5 :

Écrire un algorithme qui demande à un utilisateur de saisir un nombre pair et divisible par 3. Utiliser l'opérateur modulo pour effectuer les vérifications.

Tant que l'utilisateur se trompe, un message d'erreur explicite s'affiche à l'écran, et il doit recommencer sa saisie.

À chaque tentative, le nombre d'essais est affiché à l'écran (à l'exception de la première).

### Exercice 6 :

À partir de deux tableaux contenant l'un des prix et l'autre des quantités de produits achetés, écrire un programme permettant de calculer le prix total.

Voici les deux tableaux :

Prix : 5, 50, 23, 11

Quantités : 10, 1, 4, 3

Pour calculer le total, il faut additionner les multiplications des prix par des quantités, la correspondance étant faite à partir des indices des tableaux, tel que le montre les exemples ci-après :

$$P[0] * Q[0] = 5 * 10 = 50.$$

$$P[1] * Q[1] = 50 * 1 = 50$$

### Exercice 7 :

Écrire un algorithme permettant à un utilisateur de choisir, dans une liste de 4 produits, celui qu'il souhaite acheter :

TV à 400€

Ordinateur à 700 €

Téléphone à 200€

Portable à 900€

Les prix sont donnés hors taxe et le taux de TVA à appliquer est de 20% (utiliser une constante).

L'utilisateur doit ensuite choisir la quantité de produits voulue

Une fois les saisies effectuées, les montants HT et TTC sont affichés à l'écran.

Ci-dessous, un exemple d'affichage produit par l'algorithme programmé et exécuté :

```
Voici la liste des produits:
1 - TV : 400€
2 - Ordinateur : 700€
3 - Téléphone : 200€
4 - Portable : 900€
Quel produit voulez-vous acheter? 6
erreur de saisie
Quel produit voulez-vous acheter? 3
Combien de produits voulez-vous? 4
Total HT : 800 €
Total TTC : 960.0 €
```

### Exercice 7 bis:

On reprend la problématique de l'exercice 7, mais cette fois-ci, les produits et prix seront stockés sous forme de liste afin de pouvoir optimiser l'algorithme.

```
Liste = [{"TV", 400}, {"Ordinateur", 700}, {"Téléphone", 200}, {"Portable", 900}]
```

Rédiger votre algorithme en utilisant une boucle POUR pour balayer la liste.

On doit pouvoir ajouter un produit à notre liste sans avoir à modifier le reste de notre algo.

On rendra notre algo plus facilement modifiable en cas d'ajout de produits et plus souple.

### Exercice 8 :

Écrire un algorithme permettant de calculer et d'afficher l'aire ou le périmètre d'un rectangle après saisie au clavier d'une longueur et d'une largeur.

Vous devez créer 3 fonctions :

La première permettant d'afficher le menu.

La deuxième permettant de calculer l'aire.

Et la troisième permettant de calculer le périmètre.

L'utilisateur doit saisir la valeur 1 ou 2, correspondant au calcul qu'il veut voir affiché (aire ou périmètre).

S'il ne saisit pas l'une de ces deux valeurs, l'algorithme lui demande de recommencer

Ci-dessous, un exemple d'affichage produit par l'algorithme programmé et exécuté :

```
Quel est la longueur du rectangle? 2
Quel est la largeur du rectangle? 3
-----
-----MENU-----
1 : Périmètre
2 : Aire
-----
Quel est votre choix? 4
Vous devez choisir entre 1 ou 2
Quel est votre choix? 2
Le périmètre du rectangle est de : 10
```

### Exercice 9 :

Écrire un algorithme qui demande à un utilisateur la taille d'un carré qu'il souhaite dessiner à l'écran.

L'algorithme affiche ensuite le carré en fonction de la valeur saisie.

Aide : la hauteur d'une ligne correspond à deux fois la taille d'un caractère.

Ci-dessous, des exemples d'affichage produits par l'algorithme programmé et exécuté :



```

-----
-----MENU-----
Quelle figure voulez-vous tracer?
1 : Carré
2 : Triangle rectangle
3 : Sapin
-----
Quel est votre choix? 5
Quel est la hauteur de votre figure ? 2
Vous devez choisir entre 1, 2 ou 3
Quel est votre choix? 1
Quel est la hauteur de votre figure ? 5
*****
**      **
**      **
**      **
*****

```

**Exercice 13 :**

Écrire un algorithme qui demande à un utilisateur une phrase à convertir en Morse.

Nous ne gérons pas les chiffres dans cet algorithme, si la personne rentre en texte en minuscule, vous devrez le repasser en majuscule.

Les espaces de la phrase seront convertis en espace / espace afin de faciliter la lecture, et les caractères seront espacés de 2 espaces dans le même but.

Ci-dessous, un exemple d'affichage produit par l'algorithme programmé et exécuté :

```

Saisir une phrase à convertir en Morse: vive les bts sio

En Morse, la phrase:  VIVE LES BTS SIO  s'écrit:   ...-
..  ...-  . /  .-..  .  ... /  -...  -  ... /  ... ..
---
```

**Remarque :**

Vous pourrez utiliser le code ASCII pour faciliter les choses (la lettre A est codé 65, la lettre B est codée 66, ...)

**Exercice 14 :**

Écrire un algorithme qui demande à un utilisateur de saisir un nombre entier en base 10 et qui convertit ce nombre en en nombre binaire, après sa saisie par un utilisateur.

Exemples :

- 2 en base 10 donne 10 en binaire
- 5 en base 10 donne 101 en binaire
- 25 en base 10 donne 11001 en binaire

**Exercice 15 :**

À l'aide des algorithmes des exercices 3 et 14, créer une fonction **binaire** permettant de convertir un nombre entier en base 10 en nombre binaire, et une fonction **decimal** permettant de convertir un nombre binaire en base 10.

De plus, une fonction **menu** s'occupera de récupérer le choix de l'utilisateur (avec la prise en compte d'erreur de saisies de sa part, et aussi du nombre à convertir

Ci-dessous, un exemple d'affichage produit par l'algorithme programmé et exécuté :

<pre> ----- -----MENU----- base 10 -&gt; binaire: choix 1 binaire -&gt; base 10: choix 2 ----- Quel est votre choix? 1 Quel nombre voulez-vous convertir? 25 En binaire, 25 s'écrit : 11001 </pre>	<pre> ----- -----MENU----- base 10 -&gt; binaire: choix 1 binaire -&gt; base 10: choix 2 ----- Quel est votre choix? 3 Vous devez choisir entre 1 ou 2 Quel est votre choix? 2 Quel nombre voulez-vous convertir? 1100 En base 10, 1100 s'écrit : 12 </pre>
--	---

Si vous avez du temps, vous pouvez améliorer la fonction **menu** afin qu'elle vérifie si le nombre saisi pour le choix 2 est bien un nombre binaire.

### Exercice 16 :

Écrire une fonction nommée **bissextiles** qui prend en argument une année, et retourne un booléen : True si l'année est bissextile, False, sinon.

Voici comment vérifier qu'une année est bissextile :

« Toute année uniformément divisible par 4 est une année bissextile. Par exemple, 1988, 1992 et 1996 sont des années bissextiles.

Cependant, il y a encore une petite erreur qui doit être prise en compte. Pour éliminer cette erreur, le calendrier grégorien stipule qu'une année qui est uniformément divisible par 100 (par exemple, 1900) n'est une année bissextile que si elle est également divisible par 400.

C'est pour cette raison que les années suivantes ne sont pas des années bissextiles :

1700, 1800, 1900, 2100, 2200, 2300, 2500, 2600

En effet, elles sont divisibles par 100, mais pas par 400.

Les années suivantes sont des années bissextiles : 1600, 2000, 2400

En effet, elles sont divisibles à la fois par 100 et par 400. »

source : <https://learn.microsoft.com/fr-fr/office/troubleshoot/excel/determine-a-leap-year>

Vous pourrez tester votre fonction avec les années données dans l'énoncé.

### Exercice 17 :

Écrire un algorithme permettant à l'utilisateur de réviser les 4 opérations de bases (addition, soustraction, multiplication et division).

Deux nombres compris entre 1 et 100 sont générés aléatoirement (pour l'addition, la soustraction et la multiplication) et le choix de l'opération est fait de même.

Pour la division, il faudra que la réponse soit obligatoirement un nombre entier compris entre 1 et 20, le diviseur étant lui aussi choisi au hasard et compris entre 1 et 20

L'utilisateur doit trouver et saisir au clavier le bon résultat. Il recommence la saisie tant qu'elle n'est pas correcte.

Ci-dessous, un exemple d'affichage produits par l'algorithme programmé et exécuté :

Effectuer mentalement l'opération suivante : 82 + 41 = ? Votre réponse : <b>120</b> Réponse fausse, corrigez-vous : <b>123</b> Bonne réponse, félicitations !!!	Effectuer mentalement l'opération suivante : 133 / 19 = ? Votre réponse : <b>6</b> Réponse fausse, corrigez-vous : <b>7</b> Bonne réponse, félicitations !!!
---	--

Vous pouvez simplifier cet exercice en ne proposant que l'addition, la soustraction et la multiplication, avec un choix de nombres aléatoire compris entre 1 et 100.