

Programmation PHP - Algorithmique

1. Notes préliminaires

Vous trouverez ci-dessous un ensemble d'énoncés pour lesquels il vous faudra fournir au moins un programme PHP.

Il existe pour chaque énoncé plusieurs solutions envisageables : à vous de développer votre réflexion afin de proposer quelques-unes des ces solutions (certaines seront plus ou moins claires, d'autres plus ou moins efficaces).

2. Pythagore

Ecrire le programme qui calcule la longueur a de l'hypoténuse d'un triangle ABC rectangle en A (cf. Fig. 1).

Les longueurs AB et AC sont connues et on dispose du théorème de Pythagore : $BC^2 = AB^2 + AC^2$.

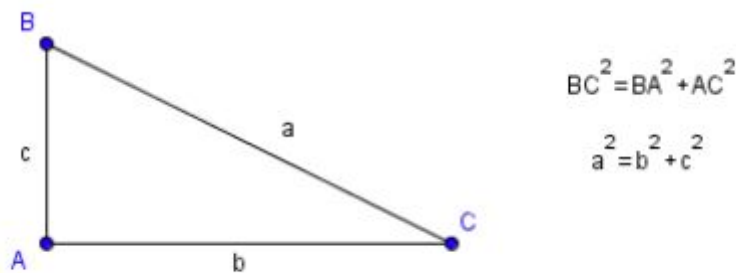


Fig. 1 : Triangle ABC rectangle en A

3. Sinus et cosinus

À partir de l'observation de l'illustration qui suit (cf. Fig. 2), écrire le programme qui calcule la position en x et y du point P en fonction de l'angle α , sachant que :

- $x = \cos(\alpha)$
- $y = \sin(\alpha)$

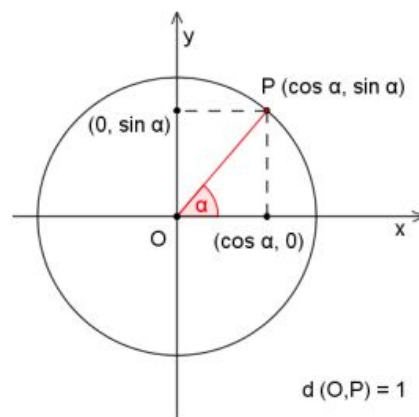


Fig. 2 : Cercle trigonométrique

4. Boucle for{...}

Exemple d'utilisation d'une boucle for {...} afin d'afficher dans l'ordre croissant les 25 premiers entiers (dans cet algorithme, nous incluons la valeur 0)

5. Boucle for {...}

Exemple d'utilisation d'une boucle for {...} afin d'afficher dans l'ordre décroissant les 25 premiers entiers (dans cet algorithme, nous n'incluons pas la valeur 0).

6. Boucle for {...}

Exemple d'utilisation d'une boucle for {...} afin d'afficher le motif suivant :

```
1
1 2
1 2 3
1 2 3 4
:
1 2 3 4 .. 25
```

7. Somme multiple

Ecrire le programme qui calcule la somme des 30 premiers entiers (privés de l'entier 0), c'est-à-dire $1+2+3+4+\dots+28+29+30$.

8. Entier pair ?

1. Ecrire le programme qui détermine si un nombre entier est pair ou non.
2. Transformer ce programme en fonction : EstPair(\$nombre).

9. Somme des entiers pairs

Variante de l'exercice 7, on souhaite additionner uniquement les 30 premiers entiers pairs (privés de l'entier 0).

Il faudra utiliser la fonction EstPair(\$nombre) qui a la valeur true si l'entier \$nombre est pair, et false s'il est impair.

10. Factorielle (n!)

Ecrire le programme qui calcule la valeur factorielle d'un nombre. Par exemple, $5! = 5*4*3*2*1$ correspond à la valeur factorielle du nombre 5. Le principe est le même que celui de l'exercice 5.

11. Echange de valeurs

Ecrire le programme qui échange les valeurs de deux variables \$a et \$b.

12. Conversion heures:minutes:secondes

Ecrire le programme qui convertit un nombre quelconque de secondes au format hh:mm:ss. Par exemple 347872s équivaut à 96h37m52s.

Ici, il faudra utiliser des opérateurs arithmétiques spécifiques (division euclidienne).

13. Affichage des éléments d'un tableau

L'objectif est d'écrire un programme qui affiche les valeurs contenues dans un tableau de chaînes de caractères de taille 6.

14. Recherche de la plus petite valeur dans un tableau d'entiers

Ecrire le programme qui recherche la plus petite valeur dans un tableau de dix entiers.

15. Tri d'un tableau d'entiers

On cherche le programme qui permet de trier par ordre croissant les valeurs d'un tableau d'entiers quelconques. Ce programme s'inspire en partie sur la recherche du plus petit élément d'un tableau d'entiers (cf. exercice 14).

16. Jeu : nombre mystère

Il s'agit d'écrire un programme qui permette à un utilisateur de deviner un nombre compris entre 0 et 1000. Le nombre à deviner est généré automatiquement par la fonction PHP existante rand(\$min, \$max) qui calcule un nombre aléatoire compris entre \$min et \$max.

Le joueur propose un nombre. Si le nombre proposé est plus petit (resp. plus grand) que le nombre à deviner, le programme devra afficher un message d'erreur du type « Le nombre que vous proposez est trop petit » (resp. « ... trop grand »). Tant que l'utilisateur n'aura pas trouvé le bon nombre, le programme lui demandera d'entrer une nouvelle valeur.