

Séance du 23.01. - Applications Boucles Gr A

Exercice 32.

Un compte informatique de Bob est protégé par le mot de passe "James-Bob". Écrire en pseudo-code puis en Python un algorithme qui demande de saisir un mot de passe jusqu'à ce que le bon mot de passe soit saisi.

Exercice 33.

Écrire en pseudo-code puis en Python un algorithme qui :

- simule les tirages successifs d'un dé jusqu'à ce qu'un 2 apparaisse,
- affiche le nombre de lancers effectués pour obtenir le 2.

Exercice 34.

Dans la mare qui est dans son jardin, Bob a mis un nénuphar. Au début, la surface du nénuphar est de 9 cm². Tous les mois, la surface du nénuphar augmente de 20%.

- Écrire une fonction **seuil** prenant en paramètre une surface de mare, et retournant le nombre de mois mis par le nénuphar pour dépasser cette surface.
- Tester la fonction **seuil** sachant que la surface de la mare est de 6 m² (c'est-à-dire 60 000 cm²).

Exercice 35.

Bob se lance dans la vente de plants de nénuphars. Il en vend 3 le premier jours. Le nombre de plants vendus double ensuite chaque jour.

- Écrire en pseudo-code puis en Python un algorithme affichant le nombre de plants vendus chaque jour jusqu'au trentième jour.
- Modifier l'algorithme précédent pour calculer le nombre total de plants vendus en 30 jours.

Exercice 36.

- Écrire en pseudo-code puis en Python une **procédure** prenant en paramètre un entier, et affichant la liste des diviseurs de cet entier.
- Modifier la procédure précédente en une **fonction** retournant le nombre de diviseurs de l'entier choisi comme paramètre.

Exercice 37 (Le jeu du nombre à deviner).

Le but du jeu est de deviner un nombre entier entre 0 et 50 choisi aléatoirement par l'ordinateur. Écrire en pseudo-code puis en Python un algorithme qui :

- tire au hasard un entier entre 0 et 50 qui sera le nombre cible,
- répète les étapes suivantes jusqu'à ce que le nombre cible soit deviné :
 - demander de saisir un entier entre 0 et 50
 - afficher un message disant si l'entier saisi est égal plus petit ou plus grand que le nombre cible
- affiche un message de victoire.

Exercice 38 (Jeu du 1 fatal).

On veut jouer à un jeu à deux joueurs dont les règles sont les suivantes :

- Le joueur 1 commence.
- Lorsque c'est son tour, chaque joueur lance un dé. Il peut relancer son dé autant qu'il veut jusqu'à 9 fois (10 lancers en tout), mais si il tombe sur un 1, il doit s'arrêter.
Le joueur gagne alors :
 - 0 points si il est tombé sur un 1,
 - la somme de ses dés sinon.
- Le premier joueur à avoir 100 points ou plus gagne.

1) Écrire en Python une fonction `tour` qui :

- ne prend pas de paramètre,
- simule un tour de jeu, en respectant les spécifications suivantes :
 - À chaque lancer de dé, la valeur du dé doit être affichée,
 - Si le dé tombe sur 1 ou si 10 lancers ont été effectués, le tour prend fin.
 - Sinon, après le lancer, le programme doit demander au joueur si il veut relancer.
Le joueur devra alors saisir 0 pour arrêter et 1 pour relancer.
- Affiche puis retourne le nombre de points gagnés au cours du tour.

Indice : on pourra utiliser une variable booléenne `continuer` pour le test de boucle.

2) Écrire en Python l'algorithme du jeu. Cet algorithme devra :

- initialiser les scores des deux joueurs à 0,
- répéter jusqu'à ce que l'un des joueurs ait atteint ou dépassé 100 points :
 - afficher le score des deux joueurs,
 - afficher un message indiquant que c'est au joueur 1 de jouer,
 - faire jouer le joueur 1 grâce à la fonction `tour`
 - si le joueur 1 n'a pas atteint 100 points, recommencer les trois étapes précédentes pour le joueur 2
- Afficher les scores finaux et le gagnant.

Pour rendre le programme plus lisible, on pourra afficher des lignes vides autant que nécessaire.