Bing

Bien sûr, voici un autre QCM de 5 questions, de type output code, en pseudo code, qui porte sur les notions clés du chapitre. Pour chaque question, choisissez la bonne réponse parmi les propositions et lisez le feedback d'expert.

Question 1:

Quel est le résultat affiché par l'algorithme suivant?

[pseudo]

fonction puissance(x : entier, n : entier) : entier

var p : entier

p ← 1

pour i de 1 à n faire

p ← p \* x

fin pour

retourner p

fin fonction

var a : entier

var b : entier

a ← 2

b ← 3

écrire(puissance(a, b))

a) 6

b) 8

c) 9

d) 27

Réponse: d) 27

Feedback: Cette question teste votre compréhension du passage des paramètres par valeur et des boucles pour. Lorsque vous appelez la fonction puissance avec les paramètres effectifs a et b, vous créez des copies des valeurs de a et b dans les paramètres formels x et n. La fonction calcule la puissance x^n en multipliant x par lui-même n fois dans une boucle pour. La fonction retourne le résultat, qui est affecté à la variable p dans le programme principal. Donc, la fonction affiche la puissance 2^3, qui est 27.

Question 2:

Quel est le résultat affiché par l'algorithme suivant?

[pseudo]

procédure afficher\_tab(tab : tableau[1..5] d'entiers)

pour i de 1 à 5 faire

écrire(tab[i])

fin pour

fin procédure

var t : tableau[1..5] d'entiers

t[1] ← 10

t[2] ← 20

t[3] ← 30

t[4] ← 40

t[5] ← 50

afficher\_tab(t)

a) 10 20 30 40 50

b) Erreur: le tableau n'est pas passé en paramètre par référence

c) Erreur: le tableau n'a pas de taille définie dans la procédure

d) Rien

Réponse: a) 10 20 30 40 50

Feedback: Cette question teste votre compréhension du passage des paramètres de type tableau. Lorsque vous appelez la procédure afficher\_tab avec le paramètre effectif t, vous passez l'adresse du premier élément du tableau t au paramètre formel tab. La procédure parcourt le tableau tab dans une boucle pour et affiche ses éléments. Donc, la procédure affiche les éléments du tableau t, qui sont 10, 20, 30, 40 et 50.

Question 3:

Quel est le résultat affiché par l'algorithme suivant?

[pseudo]

fonction somme(x : entier, y : entier) : entier

retourner x + y

fin fonction

var a : entier

var b : entier

a ← somme(2, 3)

b ← somme(a, a)

écrire(b)

a) 5

b) 8

c) 10

d) Erreur: la fonction somme est appelée deux fois

Réponse: c) 10

Feedback: Cette question teste votre compréhension du passage des paramètres par valeur et de l'appel de fonctions. Lorsque vous appelez la fonction somme avec les paramètres effectifs 2 et 3, vous créez des copies des valeurs de 2 et 3 dans les paramètres formels x et y. La fonction retourne la somme de x et y, qui est affectée à la variable a dans le programme principal. Donc, la valeur de a devient 5. Ensuite, vous appelez à nouveau la fonction somme avec les paramètres effectifs a et a, qui sont des copies de la valeur de a. La fonction retourne la somme de a et a, qui est affectée à la variable b dans le programme principal. Donc, la valeur de b devient 10 et c'est ce qui est affiché.

Question 4:

Quel est le résultat affiché par l'algorithme suivant?

[pseudo]

procédure modifier(var x : entier)

x ← x \* x - 1

fin procédure

var a : entier

var b : entier

a ← 2

b ← 3

modifier(a)

modifier(b)

écrire(a, b)

a) 2 3

b) 3 8

c) 4 9

d) 3 9

Réponse: d) 3 9

Feedback: Cette question teste votre compréhension du passage des paramètres par référence et de la modification des variables. Lorsque vous appelez la procédure modifier avec le paramètre effectif a, vous passez l'adresse de la variable a au paramètre formel x. La procédure modifie la valeur de x, qui est la même que celle de a, en lui affectant le carré de x moins 1. Donc, la valeur de a devient 3. De même, lorsque vous appelez la procédure modifier avec le paramètre effectif b, vous passez l'adresse de la variable b au paramètre formel x. La procédure modifie la valeur de x, qui est la même que celle de b, en lui affectant le carré de x moins 1. Donc, la valeur de b devient 9. Finalement, l'algorithme affiche les valeurs de a et b, qui sont 3 et 9.

Question 5:

Quel est le résultat affiché par l'algorithme suivant?

[pseudo]

fonction fibonacci(n : entier) : entier

si n = 0 ou n = 1 alors

retourner n

sinon

retourner fibonacci(n - 1) + fibonacci(n - 2)

fin si

fin fonction

var k : entier

k ← 5

écrire(fibonacci(k))

a) 3

b) 4

c) 5

d) Erreur: la fonction fibonacci est récursive

Réponse: c) 5

Feedback: Cette question teste votre compréhension du passage des paramètres par valeur et de la récursivité des fonctions. Lorsque vous appelez la fonction fibonacci avec le paramètre effectif k, vous créez une copie de la valeur de k dans le paramètre formel n. La fonction appelle elle-même avec des valeurs décroissantes de n jusqu'à atteindre les cas de base n = 0 ou n = 1. La fonction retourne alors la somme des valeurs successives de n, qui est le calcul de la suite de Fibonacci. Donc, la fonction affiche le cinquième terme de la suite de Fibonacci, qui est 5.