Bing

D'accord, je vais essayer de vous aider. Voici un 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]

procédure afficher(x : entier)

x ← x + 1

écrire(x)

fin procédure

var a : entier

a ← 5

afficher(a)

écrire(a)

a) 6 5

b) 6 6

c) 5 6

d) 5 5

Réponse: a) 6 5

Feedback: Cette question teste votre compréhension du passage des paramètres par valeur. Lorsque vous appelez la procédure afficher avec le paramètre effectif a, vous créez une copie de la valeur de a dans le paramètre formel x. La modification de x dans la procédure n'affecte pas la valeur de a dans le programme principal. Donc, la procédure affiche 6, mais la valeur de a reste 5.

Question 2:

Quel est le résultat affiché par l'algorithme suivant?

[pseudo]

fonction carré(x : entier) : entier

retourner x \* x

fin fonction

var a : entier

a ← 3

a ← carré(a)

écrire(a)

a) 3

b) 6

c) 9

d) 12

Réponse: c) 9

Feedback: Cette question teste votre compréhension du passage des paramètres par valeur et du retour des fonctions. Lorsque vous appelez la fonction carré avec le paramètre effectif a, vous créez une copie de la valeur de a dans le paramètre formel x. La fonction retourne le carré de x, qui est affecté à la variable a dans le programme principal. Donc, la valeur de a devient 9 et c'est ce qui est affiché.

Question 3:

Quel est le résultat affiché par l'algorithme suivant?

[pseudo]

procédure échanger(var x : entier, var y : entier)

var temp : entier

temp ← x

x ← y

y ← temp

fin procédure

var a : entier

var b : entier

a ← 2

b ← 4

échanger(a, b)

écrire(a, b)

a) 2 4

b) 4 2

c) 2 2

d) 4 4

Réponse: b) 4 2

Feedback: Cette question teste votre compréhension du passage des paramètres par référence. Lorsque vous appelez la procédure échanger avec les paramètres effectifs a et b, vous passez les adresses des variables a et b aux paramètres formels x et y. La modification de x et y dans la procédure affecte directement les valeurs de a et b dans le programme principal. Donc, la procédure échange les valeurs de a et b, et affiche 4 et 2.

Question 4:

Quel est le résultat affiché par l'algorithme suivant?

[pseudo]

var n : entier

fonction factorielle(n : entier) : entier

si n = 0 alors

retourner 1

sinon

retourner n \* factorielle(n - 1)

fin si

fin fonction

n ← 5

écrire(factorielle(n))

a) 1

b) 5

c) 24

d) 120

Réponse: d) 120

Feedback: Cette question teste votre compréhension du passage des paramètres par valeur et de la récursivité des fonctions. Lorsque vous appelez la fonction factorielle avec le paramètre effectif n, vous créez une copie de la valeur de n dans le paramètre formel n. La fonction appelle elle-même avec des valeurs décroissantes de n jusqu'à atteindre le cas de base n = 0. La fonction retourne alors le produit des valeurs successives de n, qui est le calcul de la factorielle. Donc, la fonction affiche la factorielle de 5, qui est 120.

Question 5:

Quel est le résultat affiché par l'algorithme suivant?

[pseudo]

var x : entier

procédure incrémenter()

x ← x + 1

fin procédure

x ← 0

incrémenter()

écrire(x)

a) 0

b) 1

c) Erreur

d) Rien

Réponse: b) 1

Feedback: Cette question teste votre compréhension de la notion de variable globale et de variable locale. La variable x est déclarée en dehors de toute procédure ou fonction, donc c'est une variable globale. Elle est accessible et modifiable dans tout l'algorithme, y compris dans la procédure incrémenter. Donc, la procédure incrémente la valeur de x, qui passe de 0 à 1, et c'est ce qui est affiché.