

Partie A : Questions de cours

1. Dans la portion d'algorithme suivant, A, B, C et D sont des entiers. Si A vaut 1 avant l'exécution de la première ligne, combien vaut D après l'exécution de ces 3 lignes ?

```
B ← 2 * A - 4
C ← -3 * A - 2 * B
D ← 15 * C + 21 * B / 3
```

- ☐ D vaut -1 ☐ D vaut 0
☐ D vaut 1 ☐ D vaut autre chose

2. Dans la portion de code suivante, n est un entier. Que vaut n après l'exécution de ces lignes ?

```
n ← 0
tant que n > 10 faire
  n ← n + 2
fin tant que
```

- ☐ n vaut 0 ☐ n vaut 22
☐ n vaut 20

3. Dans la portion de code suivante, m est un entier ayant pour valeur 7. Que vaut m lorsque l'exécution de ces lignes est terminée ?

```
faire
  si m modulo 2 = 1 alors
    m ← 3 * m + 1
  sinon
    m ← m / 2
fin si
Tant que m ≠ 1
```

- ☐ m vaut 0 ☐ on a une boucle infinie
☐ m vaut 1 ☐ m vaut 2

4. En programmation en langage C, quel signe utilise-t-on pour l'affectation ?

- ☐ = ☐ ==
☐ := ☐ !=

5. En programmation en langage C, quel signe utilise-t-on pour le test d'égalité ?

- ☐ = ☐ ==
☐ := ☐ !=

6. L'instruction «switch» sert à éviter des instructions :

- ☐ while... imbriquées. ☐ if... else... imbriquées.
☐ do... while imbriquées. ☐ for... imbriquées.

7. Dans une structure if...

- ☐ Les parenthèses encadrant la condition logique sont obligatoires.
☐ Le mot clé « else » est obligatoire.
☐ La condition, énoncée juste après if, est suivie d'un point virgule.

8. Où sont déclarées les fonctions?

- ☐ Après la fonction principale « main », si les prototypes sont définis avant.
☐ Au cours du programme principal.
☐ Avant le programme principal et la règle "une fonction doit être déclarée avant usage" doit être respectée.

9. Une fonction renvoie toujours une valeur. Comment cela se programme-t-il, à l'intérieur de la fonction ?

- ☐ En précédant la valeur à retourner par le mot-clé return.
☐ En affectant à l'identificateur de fonction la valeur à retourner.
☐ Le langage C renvoie toujours la dernière valeur calculée dans la fonction.

10. Lesquelles de ces définitions de fonctions sont correctes si elles sont sensées calculer « x » élevé à la puissance « N » entière. ?

- ☐ float puissance (float X, int N) ☐ puissance();
☐ puissance (float X ; int N) ☐ puissance : float ;
☐ puissance (float x, int N) float ☐ void puissance (float X, int N)

11. On considère l'appel suivant : échange(A, B). Pour qu'il y ait inversion des valeurs des variables A et B dans le programme principal, on utilise :

- ☐ Un passage par valeur.
☐ Un passage par adresse.

12. Pour accéder à la troisième case du vecteur Carte, on utilise l'instruction :

- ☐ Carte [3] ; ☐ Carte {3} ;
☐ Carte [2] ; ☐ Carte (2) ;
☐ Carte {2} ;

13. Quelle(s) déclaration(s) correspond(ent) à une matrice de N lignes et M colonnes ?

- ☐ float Identificateur[N][M] ; ☐ float Identificateur1 [M-1] Identificateur2 [N-1] ;
☐ float Identificateur [M-1][N-1] ; ☐ float Identificateur1 [N-1] Identificateur2 [M-1] ;

14. Pour accéder à la case située à la 2ème ligne et la 3ème colonne de la matrice T, quelle est la bonne syntaxe?

- ☐ T [2,1] ; ☐ T [2] [1] ; ☐ T (2, 1) ;
☐ T [1,2] ; ☐ T [1] [2] ;

15. Soient les déclarations suivantes :

```
struct LIVRE
{
  int nb_pages ;
  char langue[20], auteur[20], titre[20] ;
  float prix;
};
```

struct LIVRE L ;
Comment accéder à un champ d'un enregistrement ?

- ☐ auteur.L ☐ L : auteur ☐ livre.auteur
☐ L.auteur

16. On considère la déclaration suivante :

```
#define MAX 50
struct TIMBRE {
  int prix ;
  char origine[20] ;
  int annee ;
  char image[20] ;
  char couleur[20];
};
```

struct TIMBRE COLLECTION [MAX]; // Une collection est un tableau de timbres

Comment accède-t-on à l'année du 3ème timbre de la collection?

- ☐ COLLECTION [2, 2] ☐ TIMBRE[3].annee
☐ COLLECTION [2].annee ☐ COLLECTION.annee [2]
☐ COLLECTION [2, annee] ☐ COLLECTION.annee