

QCM de langage C

Question 1

Qu'affichent les instructions suivantes :

```
char a ;  
a=65 ;  
printf("%d %c",a,a) ;
```

- | | |
|---------|----------|
| 1) A A | 3) A 65 |
| 2) 65 A | 4) 65 65 |
-

Question 2

Qu'affiche l'instruction suivante :

```
printf("j'aime le C %d fois", 30/20) ;
```

- | | |
|-----------------------|-------------------------|
| 1) j'aime le C 0 fois | 3) j'aime le C 1.5 fois |
| 2) j'aime le C 1 fois | 4) je n'aime pas le C |
-

Question 3

Qu'affichent les instructions suivantes :

```
int x=9 ;  
int y=x+10 ;  
printf("%d :%d :%d",x,y,y) ;
```

- | | |
|--------------|---------------|
| 1) 9 :19 : | 3) 9 :19 :%d |
| 2) 9 :19 :19 | 4) %d :%d :%d |
-

Question 4

Qu'affichent les instructions suivantes :

```
int i=10 ;  
while (i>0) {  
    i=i-4 ;  
    printf("%d ",i) ;  
}
```

- | | |
|------------|-----------------------------|
| 1) 1 0 6 2 | 3) 6 2 |
| 2) 6 2 -2 | 4) C'est une boucle infinie |
-

Question 5

Quelle est la valeur de S après exécution des instructions suivantes :

```
int i ;  
int S=0 ;  
for (i=1 ; i<6 ;i=i+2)
```

$S = S + i$;

1) 6

2) 7

3) 8

4) 9

Question 6

Le type « char » correspond à :

1) Un entier codé sur un unique octet

2) Un réel

3) Une chaîne de caractères

4) Aucune de ces trois réponses.

Question 7

Dans une structure if ...

1) Les parenthèses encadrant la condition logique sont obligatoires

2) Le mot clé « else » est obligatoire

3) La condition, énoncée juste après if, est suivie d'un point virgule.

Question 8

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

1) =

2) :=

3) ==

Question 9

Si le nombre d'itérations est connu, il est conseillé d'utiliser :

1) while ...

2) do ... while

3) for ...

Question 10

Si le nombre d'itérations n'est a priori pas connu et que l'on désire passer au moins une fois dans la boucle, on utilise :

1) while ...

2) do ... while

3) for ...

Question 11

On considère deux tableaux T1 et T2. Entourez les propositions correctes. Il est possible de copier le contenu de T2 dans T1 sans perdre d'information :

1) directement si T1 et T2 sont de même taille : on utilise l'instruction $T1 = T2$

2) directement si la taille de T1 est supérieure à la taille de T2 : on utilise l'instruction $T1 = T2$

3) directement si la taille de T2 est supérieure à la taille de T1 : on utilise l'instruction $T1 = T2$

- 4) élément par élément à l'aide d'une boucle dès que la taille de T1 est supérieure ou égale à la taille de T2.
-

Question 12

Soit un tableau `int T[10]` ; pour accéder à la troisième case du tableau T, on utilise :

- 1) `T[3]`
 - 2) `T[2]`
 - 3) `T{2}`
 - 4) `T(3)`
-

Question 13

Lesquelles de ces déclarations de fonctions sont correctes si elles sont sensées calculer « x » élevé à la puissance « n » entière ?

- 1) `double puissance (double x, int n) ;`
 - 2) `puissance (double x, int n) ;`
 - 3) `void puissance (double x, int n) ;`
 - 4) `puissance()`
-

Question 14

Quelles sont les manières correctes pour passer le tableau d'entiers `tab` en argument ?

- 1) `void fonction (int tab, int dim)`
 - 2) `void fonction (int tab[], int dim)`
 - 3) `void fonction (int &tab, int dim)`
-

Question 15

En C, on peut utiliser les variables sans pour autant avoir à les déclarer.

- 1) Vrai
 - 2) Faux
-

Question 16

Entourez les réponses correctes. Dans un fichier d'entête .h, on met :

- 1) Les définitions des fonctions
 - 2) Les entêtes ou prototypes des fonctions
 - 3) Les définitions des structures
 - 4) La fonction `main`
 - 5) Toutes les variables du projet
-

Question 17

Soit le pointeur de fichier défini par `FILE *fp` ; on veut ouvrir le fichier

« C:/Mesdocuments/essai.txt » en mode lecture. Entourez les propositions correctes.

- 1) `"C:/Mesdocuments/essai.txt" = fopen(fp, "r") ;`
- 2) `fp = fopen("C:/Mesdocuments/essai.txt", "r") ;`
- 3) `fp = fopen(C:/Mesdocuments/essai.txt, "r") ;`
- 4) `fopen(fp, "C:/Mesdocuments/essai.txt", "r") ;`
- 5) le fichier « C:/Mesdocuments/essai.txt » doit exister

6) Si le fichier «C:/Mesdocuments/essai.txt » n'existe pas, il est créé.
