

QCM - Informatique

Questions 16 à 30

16. Soit le programme C suivant :

```
#include <stdio.h>
void appel(int x, int y){x = x + 1; y = y + 2;}
int main(void){
    int x = 2;
    int y = 5;
    appel(x,y);
    printf("x=%d", x);
    printf("\n");
    printf("y=%d", y);
    return 0;
}
```

Quel est le résultat retourné lors de l'exécution de ce programme ?

- A. x=0, y=0 ?
- ☒ B. x=2, y=5 ?
- C. x=0, y=1 ?
- D. x=3, y=7 ?
- E. aucune valeur car erreur détectée lors de l'exécution.

17. Soit l'équation de récurrence suivante :

$$T(n) = 2 * T(n/2) + n, \text{ avec } T(1) = \text{constante}$$

Quelle est la valeur de $T(n)$ pour n suffisamment grand, K étant une constante ?

- A. $K \times \log n$?
- B. $K \times n$?
- C. $K \times n \log n$?
- D. $K \times n^2$?
- E. $K \times n^n$?

18. Soit la fonction :

$$f(n) = 100n^{10} + 3n^5 + 10n^2 + \frac{2^n}{100}$$

Voici le code en Python de l'implémentation de cette fonction :

```
def f(n) :
    return 100*n**10 + 3*n**5+10*n**2 + 2**n/100
```

Le code de l'implémentation de cette fonction aurait été équivalent en C ou en Java. Donnez la valeur de la complexité de cette fonction :

- A. $\mathcal{O}(1)$,
- B. $\mathcal{O}(n)$,

C. $\mathcal{O}(n^2)$,

D. ~~$\mathcal{O}(n^{10})$~~ , $\theta(n^{10})$

E. $\mathcal{O}(2^n)$.

19. Pour un ordinateur donné les nombres entiers sont stockés sur 4 octets, les nombres flottants sur 4 octets également, et les caractères sur 1 octet. Soit le programme :

```
#include <stdio.h>
union D {
    int i;
    float f;
    char s[20];
};

int main()
{
    union D d;
    printf( "Taille me'moire occupe'e par les donne'es~: %d\n", sizeof(d));
    return 0;
}
```

Quel sera le résultat affiché lors de l'exécution de ce programme ?

~~A. 20 octets ?~~

B. 30 octets ?

C. 28 octets ?

D. 32 octets ?

E. 64 octets ?

20. Un objet se déplace sur une droite graduée. Au début du processus (à l'instant $t = 0$) l'objet se trouve à la position 0.

Si, à l'instant t , l'objet est à la position p , il sera à l'instant $t + 1$ à la position :

- $p + 1$ avec une probabilité égale à $1/3$,
- $p + 3$ avec une probabilité égale à $1/3$,
- $p - 4$ avec une probabilité égale à $1/3$.

Soit l'algorithme implémentant le déplacement de l'objet pour 1000 itérations, écrit à l'aide de la fonction randow() qui renvoie un nombre pseudo aléatoire entre 0 et 1.

Début du programme

$p=0$

Pour t de 1 à 1000 Faire

Si (random())<1/3) alors

$p=p+1$

Finsi

Si (random())>=1/3) et (random())<2/3) alors

$p=p+3$

Finsi


```

Si (random())>=2/3) alors
    p=p-4
Finsi
Finpour.
Afficher p.
Fin du programme

```

Cet algorithme :

- A. ~~Implémente~~ correctement le déplacement de l'objet à chaque itération?
- B. ~~2. N'implémente pas~~ correctement le déplacement de l'objet mais il y aura toujours un seul déplacement à chaque itération?
- C. N'implémente pas correctement le déplacement de l'objet car il n'y aura pas forcément un seul déplacement à chaque itération? }
- D. N'implémente pas correctement le déplacement de l'objet car il n'y aura aucun déplacement à chaque itération?
- E. N'implémente pas correctement le déplacement de l'objet car il fait aller-retour en deux itérations consécutives?

21. Soit la fonction fct() implémentée dans le langage Python :

```

def fct(x,n):
    if n==0:
        return 1
    y=fct(x,n//2)
    if (n%2)==1:
        return y**2*x
    else:
        return y**2

```

2, 10
 $y = (2, 5) \rightarrow 16$
 $y = (2, 2) \rightarrow 8$
 $y = (2, 1) \rightarrow 2$
 $y = (2, 0) \rightarrow 1$

- 3
 - 2

Pour tout $n \geq 1$ le nombre total de multiplications effectuées par un appel à fct(x,n) est inférieur ou égal à : (on donnera uniquement la meilleure réponse)

- A. $2 \times \lfloor \log_2 n \rfloor$,
- B. $2 + 2 \times \lfloor \log_2 n \rfloor$,
- C. $n \times \lfloor \log_2 n \rfloor$,
- D. $n + n \times \lfloor \log_2 n \rfloor$,
- E. $n^2 \times \lfloor \log_2 n \rfloor$.

22. Donnez le résultat de la fonction précédente `fct()` appliquée avec les paramètres $a = 2$ et $n = 10$,

- A. 20.
- B. 200.
- C. 512.
- D. 1024.
- E. 1048576.

23. En arithmétique, un triplet pythagoricien est un triplet (x, y, z) d'entiers naturels non nuls vérifiant la relation de Pythagore : $x^2 + y^2 = z^2$. On souhaite trouver tous les triplets (x, y, z) vérifiant la condition $x + y + z \leq 50(1)$. Soit le programme Python suivant :

```
def triplets(val):  
    x,y,z=0,0,0  
    for k in range(val):  
        z=x**2 + y**2  
        if x+y+math.sqrt(z) <= val and math.sqrt(z).is_integer() :  
            print(x,y,int(math.sqrt(z)))  
        z=0  
        x=x+1
```

`triplets(50)`

L'exécution de ce programme donnera comme résultat :

- A. Aucun triplet vérifiant la condition (1)?
- B. Tous les triplets vérifiant la condition (1)?
- C. Un certain nombre de triplets vérifiant la condition (1)?
- D. Aucun résultat car le programme ne fonctionne pas?
- E. Aucune des réponses précédentes?

24. Soit le programme Java :

```
class C  
{  
    public static int x;  
    public int y;  
  
    public C(){ x++; y=x; }  
  
    public static void main (String args[] )  
    {  
        C c=new C();  
        C b= new C();  
        C a= c;  
        System.out.println(a.x + " et " + a.y);  
    }  
}
```


23. Une fois cette classe exécutée lequel de ces résultats est retourné ?

- A. 0 et 0 ?
- B. 1 et 0 ?
- C. 1 et 1 ?
- D. 0 et 1 ?
- E. 2 et 1 ?

25. Soit le programme Java suivant :

```
class A{  
    public int f(){ return(1); }  
    public static int g(){ return (2); }  
}
```

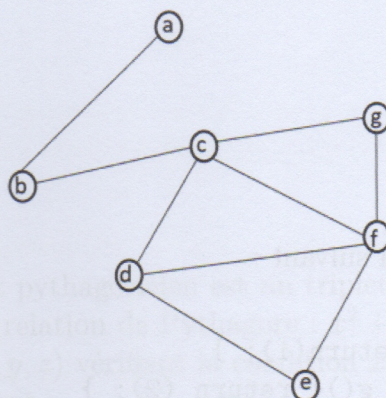
```
class B extends A{  
    public int f(){ return(3);}  
    public static int g(){ return (4); }  
}
```

```
class E{  
    public static void main (String args[] )  
    {  
        A a;  
        B b=new B();  
        a = b;  
        System.out.println(a.f() + " et " + a.g());  
    }  
}
```

Une fois cette classe exécutée lequel de ces résultats est correct ?

- A. 0 et 0 ?
- B. 1 et 0 ?
- C. 2 et 1 ?
- D. 3 et 2 ?
- E. ne compile pas ?

26. Soit le graphe G :



Indiquez quelles assertions sont vraies :

- A. Le graphe est connexe,
- B. Le graphe est complet,
- C. Le graphe est eulérien,
- D. Le graphe est hamiltonien,
- E. L'ordre de ce graphe est égal à 9.

27. Quelle est la négation de la proposition suivante ?

$$\forall \epsilon \in \mathbb{R}_+^*, \exists \eta \in \mathbb{R}_+^*, \forall x \in \mathbb{R}, \text{abs}(x - 1) \leq \eta \Rightarrow \text{abs}(x^2 - 1) \leq \epsilon$$

- A. $\exists \epsilon \in \mathbb{R}_+^*, \forall \eta \in \mathbb{R}_+, \exists x \in \mathbb{R}, \text{abs}(x - 1) > \eta \Rightarrow \text{abs}(x^2 - 1) > \epsilon$
- B. $\exists \epsilon \in \mathbb{R}_+^*, \forall \eta \in \mathbb{R}_+, \exists x \in \mathbb{R}, \text{abs}(x - 1) \leq \eta \Rightarrow \text{abs}(x^2 - 1) > \epsilon$
- C. $\exists \epsilon \in \mathbb{R}_+^*, \forall \eta \in \mathbb{R}_+, \exists x \in \mathbb{R}, \text{abs}(x - 1) \leq \eta \wedge \text{abs}(x^2 - 1) > \epsilon$
- D. $\exists \epsilon \in \mathbb{R}_+^*, \forall \eta \in \mathbb{R}_+, \exists x \in \mathbb{R}, \text{abs}(x - 1) > \eta \wedge \text{abs}(x^2 - 1) > \epsilon$
- E. $\forall \epsilon \in \mathbb{R}_+^*, \forall \eta \in \mathbb{R}_+, \exists x \in \mathbb{R}, \text{abs}(x - 1) > \eta \wedge \text{abs}(x^2 - 1) > \epsilon$

28. Soit, en le langage C, l'expression suivante :

```
int (*p)(int(*)[ ])
```

Indiquez quelle assertion est vraie :

- A. p est un pointeur vers une fonction qui prend en argument un pointeur vers un tableau d'entiers, et qui renvoie un entier,
- B. p est une fonction qui prend en argument un pointeur vers un tableau d'entiers et qui renvoie un pointeur d'entier,
- C. p est un pointeur vers une fonction qui prend en argument un tableau de pointeurs d'entiers, et qui renvoie un entier,
- D. p est une fonction qui rend un pointeur d'entier dont les arguments sont des tableaux de pointeurs d'entiers,
- E. n'a aucun sens.

29. Calculer l'enveloppe convexe d'un ensemble de points est :

- A. impossible en général, cela dépend de l'organisation de l'ensemble des points,
- B. possible en $\mathcal{O}(n \log n)$,
- C. possible mais seulement en $\mathcal{O}(n^2)$,
- D. possible mais seulement en $\mathcal{O}(n^2 \log n)$,
- E. possible mais seulement en $\mathcal{O}(n^3)$.

30. En langage C, que rend le code suivant :

```
unsigned int n=0;  
printf(''n = %d'', (n==2));
```

- A. 2 ?
- B. 1 ?
- C. 0 ?
- D. 8 ?
- E. Aucune des réponses précédentes ?

