



ECOLE DES PONTS PARISTECH, ISAE-SUPAERO,
ENSTA PARIS, TELECOM PARIS, MINES PARIS,
MINES SAINT-ETIENNE, MINES NANCY, IMT ATLANTIQUE,
ENSAE PARIS, CHIMIE PARISTECH – PSL.
ECOLE POLYTECHNIQUE, ARTS et METIERS,
ESPCI PARIS, SUOPTIQUE, ENAC.

Admission par voie universitaire

EPREUVES de SPÉCIALITÉ

Durée de l'épreuve : 2 heures.

L'emploi de tous documents (dictionnaires, imprimés, ...) et de tous appareils (traductrices, calculatrices électroniques, ...) est interdit dans cette épreuve.

Cette épreuve est un questionnaire à choix multiples.

Vous devez composer les spécialités en fonction
de vos choix au moment de l'inscription.

Questions 1 à 15 pour l'épreuve d'Electricité, Electronique et Automatique ;

Questions 16 à 30 pour l'épreuve d'Informatique ;

Questions 31 à 45 pour l'épreuve de Sciences du Vivant ;

Questions 46 à 60 pour l'épreuve de Mécanique ;

Questions 61 à 75 pour l'épreuve de Génie Civil ;

Questions 76 à 90 pour l'épreuve de Chimie.

Questions 91 à 105 pour l'épreuve de Probabilités/Statistique.

Chaque question peut admettre, de façon variable,
entre une et cinq réponses correctes.

Dans toutes les épreuves vous indiquerez les assertions correctes.

Exprimer les réponses exactes en noircissant la ou les cases correspondantes.

Toute réponse incorrecte sera pénalisée.

Les feuilles dont l'entête d'identification n'est pas entièrement
renseigné ne seront pas prises en compte pour la correction.

Respectez scrupuleusement les consignes de remplissage
des cases du document réponse.

QCM - Informatique

Questions 16 à 30

16. Soit N un entier défini en écriture binaire : 10011110.

La valeur de N en base 4 vaut :

- A. 21310.
- B. 2132.
- C. 3203.
- D. 2310.
- E. aucune valeur précédentes.

17. Soit le code suivant écrit en C :

```
int i,j,inter;
for(j=1;j<=n;j++){
for(i=0;i<n-1;i++){
if (T[i]>T[i+1]){
inter = T[i];
T[i] = T[i+1];
T[i+1] = inter;
}}}
```

Ce code réalise :

- A. un tri par sélection.
- B. un tri par insertion.
- C. un tri par permutation.
- D. un tri rapide.
- E. un tri par bulles.

18. Donner, dans le meilleur des cas, la complexité Θ du tri précédent si n est le nombre d'éléments du tableau.

- A. $\Theta(n)$.
- B. $\Theta(n^2)$.
- C. $\Theta(n^3)$.
- D. $\Theta(n \log n)$.
- E. Aucune des valeurs précédentes.

19. Soit le programme suivant écrit en langage C :

```
#include <stdio.h>
#define CARRE(v) v*v
#define CUBE(v) v*v*v
struct{int SommeCarres;int SommeCubes;}Somme

void affiche(int i,int j,int k){
Somme.SommeCarres+=j;
Somme.SommeCubes+=k;
printf("(%d,%d) ",j,k);
}

int main(void){
int i=1;
while(i<5){affiche(i++,CARRE(i),CUBE(i)); i++;} }
```

Ce programme affiche comme résultat :

- A. (1,1) (4,8) (9,27) (16,64) (25,125).
- B. (0,0) (1,1) (4,8) (9,27) (16,64).
- C. (1,1) (9,27).
- D. Aucune valeur.
- E. Le programme ne compile pas.

20. Soit le code suivant écrit en Python :

```
x = 5
x++
print(x)
```

Ce programme affiche comme résultat :

- A. 4.
- B. 5.
- C. 6.
- D. 7.
- E. Une erreur.

21. Soit le code suivant écrit en Python :

```
x = 5
++x
print(x)
```

Ce programme affiche comme résultat :

- A. 4.
- B. 5.
- C. 6.
- D. 7.
- E. Une erreur.

22. Soit le code suivant écrit en Python :

```
x = 5
x+=1
print(x)
```

Ce programme affiche comme résultat :

- A. 4.
- B. 5.
- C. 6.
- D. 7.
- E. Une erreur.

23. Soit le programme écrit en langage Java :

```
import java.util.Arrays;

public class ClassA {
    public static void main(String[] args) {
        String un = "Soleil";
        StringBuilder builder = new StringBuilder();
        for(char elem : Arrays.asList('S','o','l','e','i','l')) {builder.append(elem);}
        String deux = builder.toString();
        System.out.printf("%b , %b",un == deux,un.equals(deux));
    }
}
```

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

- A. true , true.
- B. false , false.
- C. true , false.
- D. false , true.
- E. 'Soleil'.

24. Dans un paquetage Java, constitué de plusieurs classes, quel est le mode de visibilité d'une méthode dite *friendly* ou *package friendly* appartenant à l'une des classes de ce paquetage ?
- A. Toutes les méthodes des autres classes du package courant ont accès au membre *friendly* mais pas les méthodes des classes extérieures au paquetage.
 - B. Toutes les autres classes du package courant ont accès au membre *friendly* et également toutes les classes extérieures au paquetage.
 - C. Toutes les méthodes des autres classes du package courant n'ont pas accès au membre *friendly* mais c'est possible pour toutes les méthodes des classes extérieures au paquetage.
 - D. Une méthode *friendly*, permet son utilisation seulement dans le cas d'une classe dérivée de la classe de cette méthode mais implémentée à l'extérieur du paquetage.
 - E. Ce n'est pas un mode de visibilité du langage Java pour une méthode.

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

26. Soit, en langage C, l'expression suivante :

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

Alors,

- 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. cette expression n'a aucun sens.

27. 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 $\Theta(n \log n)$.
 - C. possible mais seulement en $\Theta(n^2)$.
 - D. possible mais seulement en $\Theta(n^2 \log n)$.
 - E. possible mais seulement en $\Theta(n^3)$.

28. 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.
29. 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. $\Theta(1)$.
- B. $\Theta(n)$.
- C. $\Theta(n^2)$.
- D. $\Theta(n^{10})$.
- E. $\Theta(2^n)$.

30. 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. $K \times n^n$.

