



**Concours d'entrée en 1^{ère} année du cycle Master
Bases Informatiques / Computer science basics**

Durée : 1h30

« L'objectif n'est pas de chercher à tout prix à traiter toute l'épreuve en sprintant inconsidérément et en bâclant les raisonnements mais d'en couvrir une part significative de manière convaincante »
ND/NG

"The goal is not to try to cover the entire test at all costs by sprinting recklessly and sloppy reasoning but to cover a significant portion of it convincingly" ND/NG

Partie 1 : Questions à choix multiples / Multiple choice questions (11 points)

1 point pour chaque bonne réponse. -0.25 pour chaque mauvaise réponse. 0 point si aucune réponse.

1 point for each correct answer. -0.25 for each wrong answer. 0 point if no answer.

Pour chacune de ces questions, sélectionner la bonne réponse.

1. Soit la fonction ci-dessous

```
Fonction Calcul (n : entier) : entier
Var i: entier ;
Var S: réel ;
DébutFonction
  i = 0 ;
  S = 0.0 ;
  Pour i allant de 1 à n+1 par pas de 1 Faire
    S = S + 1/i;
  FinPour
  Retourner i;
Fin Fonction
```

Lorsqu'on appelle cette fonction avec la valeur 10, elle va retourner la valeur :

- a. 0
- b. 1
- c. 10
- d. 11
- e. 12

2. La conversion de 145 de la base décimale à la base octale donne :

- a. 217
- b. 221
- c. 234
- d. 311
- e. 331

3. Soit les fonctions suivantes

```
Fonction Calcul_un (n : entier) : réel
DébutFonction
  Si n == 0 Alors
    Retourner 0.0 ;
  FinSi
  Retourner 1/n ;
Fin Fonction
```

```
Fonction Calcul_deux (n : entier) : réel
DébutFonction
  Si n == 0 Alors
    Retourner 0.0 ;
  Sinon
    Retourner 1/n ;
  FinSi
Fin Fonction
```



On peut dire qu'elles sont équivalentes

- a. Vrai
 - b. Faux
 - c. Ça dépend de n
4. Soit fait la fonction ci-contre

```
Fonction Calcul(p : entier) : entier
  Var i, k : entier ;
  DébutFonction
    Si p == 1 Alors
      Retourner 1 ;
    FinSi
    Pour i allant de 0 à p-1 par pas de 1 Faire
      K = K * i ;
    FinPour
    Retourner K ;
  Fin Fonction
```

Cette fonction permet de calculer ?

- a. $(p - 1)!$
 - b. $p!$
 - c. p^{i-1}
 - d. p^i
 - e. Aucune des réponses ci-dessus
5. Soit la fonction Traitement ci-dessous :

```
Fonction Traitement (T : tableau de 54 entier) : booléen
  Var i, n: entier ;
  DébutFonction
    i = 0 ;
    n = longueur(T) ;
    TantQue T[i] != T[i+1] et i <= n-2 Faire
      i = i+1 ;
    FinTantQue
    Retourner i == (n-1);
  Fin Fonction
```

Elle permet de

- a. Compter les éléments du tableau T
 - b. Dire si oui ou non les éléments du tableau sont tous distincts
 - c. Dire si oui ou non les éléments consécutifs du tableau sont distincts
 - d. Permuter les éléments du tableau dans un ordre quelconque
6. Soit l'instruction $j = ++i++$; Sachant qu'avant l'exécution de cette i valait 5.
Après l'exécution de cette instruction j vaut ?
- a. 5 ;
 - b. 6 ;
 - c. 7 ;
 - d. On ne peut pas déterminer ;



7. Soit l'instruction `j = ++i++` ; Sachant qu'avant l'exécution de cette `i` valait 5.
Après l'exécution de cette instruction `i` vaut ?

a. 5 ;
b. 6 ;
c. 7 ;
d. On ne peut pas déterminer ;

8. Soit la fonction maths ci-dessous.

```
Fonction maths (x :réel) : réel  
Var a, b: réel ;  
DébutFonction  
  a = x * x ;  
  b = a / x - 1;  
  b = a - b + 3  
  Retourner b ;  
Fin Fonction
```

Elle permet de calculer (pour x non nul):

a. $x^2 - x - 1$
b. $x(x - 1) + 4$
c. $x^2 - \frac{x^2}{x-1} + 3$
d. $x^2 - x + 2$

9. Lequel de ces noms de variables est correct ?

a. `une_deux`
b. `fichier.txt`
c. `i0`
d. `_tired`

10. Qu'affiche le programme suivant ?

```
void compute(int x) {  
  int x = 50;  
  { int y = x; y = 20; }  
  printf("%d", y);  
}  
  
main() {  
  compute (101) ;  
}
```

a. 20
b. 50
c. 101
d. Rien

11. En C, un type se construit avec:

a. `enum`
b. `union`
c. `struct`
d. `typedef`



Partie 2 : Écriture de quelques algorithmes (9 points)

Exercice 1 : 4 points

Écrire une fonction qui pour un entier n donné, retourne si oui ou non l'entier est une puissance de 7.

Exercice 2 : 5 points

Soit l'enregistrement Élève comportant les champs suivants : Nom, classe, noteS1 et noteS2. noteS1 et noteS2 représentent respectivement la moyenne de l'élève au semestre 1 et au semestre 2 de l'année scolaire 2020-2021.

- Écrire une fonction qui prend en paramètre un tableau de 91 Élève et retourne la moyenne des élèves de la classe Licence 1
- Écrire une fonction qui prend en paramètre un tableau de 91 Élève et retourne le nombre d'élèves ayant baissé de plus de 10% entre les 2 semestres

Partie 3 : Langage SQL (10 points)

Soit le schéma relationnel suivant :

- Etudiant (matricule, nomComplet, dateNaissance, sexe, lycee, moyBac, #classe, #residence)
- Classe (niveau, filiere, idClasse)
- Adresse (idAdresse, quartier, repere, ville)
- Bus (idBus, surnom)
- PointDeRamassage (surnom, #position, #idBus)
- RamassageDUnEtudiant (#matricule, #pdr, dateDebut, dateFin, paye)
- Stats(id, message)

Notes :

- L'attribut classe de Etudiant fait référence à idClasse de la relation Classe
- L'attribut residence de Etudiant fait référence à idAdresse de la relation Adresse
- Position de PointDeRamassage fait référence à idAdresse de la relation Adresse
- idBus de PointDeRamassage fait référence à idBus de la relation Bus
- matricule de RamassageDUnEtudiant fait référence à matricule de la relation Etudiant
- pdr de RamassageDUnEtudiant fait référence à surnom de PointDeRamassage

1. Que fait la requête suivante ? **2 points**

```
SELECT COUNT(*) FROM Etudiant WHERE moyBac >= 14.0 and lycee = "Mendong";
```

2. Que fait la requête suivante ? **2 points**

```
SELECT DISTINCT niveau FROM Classe WHERE filiere = "Ingénieur" ORDER BY niveau;
```



3. Que fait la requête suivante ? **2 points**

```
SELECT idClasse, COUNT(*) as nb FROM Etudiant e, Classe c
WHERE e.classe = c.idClasse
GROUP BY idClasse HAVING nb >= 15;
```

4. On suppose que la table Etudiant contient les informations suivantes :

matricule	nomComplet	sexe	dateNaissance	moyBac	lycee	classe	residence
1918i222	Mopao Cyril	M	2002-10-10	10.00	Biyem Assi	3ISI	AD1
1918i223	Toko Fabrice	M	2004-12-11	14.00	Emana	3ISI	AD5
1918i002	Dipa Mispas	F	2003-01-11	12.22	Essos	3SRT	AD12
1918i007	Moko André	M	1999-04-03	10.65	College Laval	3SRT	AD7
1918i009	Komo Momo	M	2005-03-11	10.65	College Laval	3SRT	AD4
1918i200	Baza Ulrich	M	2001-05-03	10.65	College Vogt	3SRT	AD15
1516i205	Moko André	M	1999-04-03	10.65	College F X Vogt	4ISI	AD7
1516i277	Moki Fred	M	1999-08-03	10.65	Jean Tabi	4ISI	AD16
1918i202	Mopao Steve	M	2000-10-10	11.65	Biyem Assi	4ISI	AD2
1918i204	Goko Cedric	M	2001-03-03	10.65	College Vogt	4ISI	AD3
1617i003	Kiki Marie	M	2005-03-11	12.22	Mendong	4SRT	AD10
1718i012	Toki Kozaki	F	2003-01-11	16.11	Essos	4SRT	AD11
1718i013	Toki Kozo	F	2003-12-11	14.11	Retraite	4SRT	AD14
1918i012	Koko Francine	F	2003-01-11	13.11	Essos	4SRT	AD9

a. Donner le résultat de l'exécution de la requête (**2 points**)

```
SELECT matricule, nomComplet from Etudiant WHERE matricule NOT LIKE "1918%" AND
classe = "4SRT" ORDER by nomComplet;
```

b. Donner le résultat de l'exécution de la requête (**2 points**)

```
SELECT matricule, nomComplet, moyBac from Etudiant WHERE YEAR(dateNaissance) >=2000
AND moyBac > 12;
```