



Département d'informatique

**EPREUVE
D'INFORMATIQUE**
MINI SESSION N° 2
Janvier 2025

Classe : P C-D

Durée : 2h

Coef :

Exercice 1 (8 pts)

Le directeur de l'OBC (Office du Baccalauréat du Cameroun) souhaite améliorer le cryptage des données échangées sur la plateforme de gestion des examens de ladite institution. Vous êtes sollicité pour l'écriture d'un algorithme ou clé de cryptage qui permettra de résoudre ce problème. Le cryptage d'une expression à l'aide de cet algorithme sera fait de la manière suivante : le 1^{er} caractère de l'expression sera permute avec le dernier caractère, le 2^{ème} sera permuté avec l'avant dernier et ainsi de suite. Exemple : le cryptage de l'expression **mini** donnera **inim** et celui de **mange** donnera **egnam**.

- 1) Proposer une structure de données qui permettra de conserver en mémoire l'expression qui sera cryptée, sachant qu'une expression aura au maximum 50 caractères. **0.5 pt**
- 2) Donner un avantage de l'utilisation de cette structure de données. **0.5 pt**
- 3) Citer deux autres structures de données. **0.25*2=0.5 pt**
- 4) Déclarer une variable **expr** qui permettra de conserver l'expression à crypter. **1 pt**
- 5) Proposer une structure de contrôle qui sera utilisée dans l'algorithme de cryptage. **0.5 pt**
- 6) Écrire l'algorithme de cryptage sachant qu'il permettra d'effectuer les tâches suivantes :
 - Entrer tous les caractères de l'expression à crypter. **1 pt**
 - Chiffrer l'expression. **3 pts**
 - Afficher l'expression cryptée. **1 pt**

Exercice 2 (12 pts)

On considère l'algorithme ci-contre, écrit par un élève de première D dans le cadre d'un projet informatique. Il souhaite le traduire en C et l'exécuter à travers un IDE. Mais dans un premier temps, il voudrait le tester en l'exécutant lui-même.

- 1) Exécuter à la main cet algorithme si l'utilisateur entre successivement les notes suivantes : 12 ; 9 ; 15 ; 13. **2 pts**
- 2) Réécrire les lignes 8 à 11 à l'aide de la structure *tant que*. **1.5 pt**
- 3) Donner le rôle d'un IDE. **0.75 pt**
- 4) Proposer un IDE que cet élève pourra utiliser. **0.25 pt**
- 5) Spécifier la bibliothèque C à utiliser dans la traduction de cet algorithme. **0.5 pt**

1. **Algorithme Gestion**
2. **var i:entier;**
3. **mg, S: réel;**
4. **Notes: Tableau[4] de réel;**
5. **Début**
6. **écrire ("entrer les notes des 4 élèves ");**
7. **S←0;**
8. **pour i allant de 1 à 4 faire**
9. **lire(Notes[i]);**
10. **S← S+Notes[i];**
11. **finpour**
12. **mg←S/4;**
13. **écrire(mg);**
14. **Fin**

- 6) Justifier l'utilisation d'une telle bibliothèque. 1 pt
- 7) Déclarer la variable **Notes** en C. 0.5 pt
- 8) Traduire les lignes 8 à 11 de cet algorithme en C. 1.5 pt
- 9) Cet élève décide plutôt d'utiliser une procédure dans cet algorithme. Il obtient alors l'algorithme suivant :

```

1. Algorithme Gestion
2. procédure MoyGene( T : tableau[4] de réel)
3.   var j : entier ;
4.   M, S : réel ;
5.   début
6.   S $\leftarrow$ 0 ;
7.   Pour j allant de 1 à 4 faire
8.     S $\leftarrow$ S+ T[j] ;
9.   Finpour
10.  M $\leftarrow$ S/4 ;
11.  Ecrire(M) ;
12. Finprocédure
13. var i:entier;
14.   Notes: Tableau[4] de réel;
15. Début
16.   écrire ("entrer les notes des 4 élèves ");
17.   pour i allant de 1 à 4 faire
18.     lire(Notes[i]);
19.   finpour
20.   MoyGene(Notes) ;
21. Fin

```

- a) Définir : procédure. 0.5 pt
- b) Donner un avantage de l'utilisation d'une procédure dans cet algorithme. 0.5 pt
- c) Relever dans cet algorithme
- Les lignes de déclaration de la procédure. 0.5 pt
 - La ligne d'appel de la procédure. 0.5 pt
 - Un paramètre effectif et un paramètre formel. 0.5*2=1 pt
 - Une variable locale et une variable globale. 0.5*2=1 pt