

COLLEGE Mgr François-Xavier VOGT		Année scolaire 2023-2024
DÉPARTEMENT D'INFORMATIQUE		Date : 31 janvier 2024
MINI-SESSION ÉPREUVE D'INFORMATIQUE		
Classe : Premières CD		Durée : 02 heures

Aucun document ou matériel en dehors de ceux remis aux candidats par l'examineur n'est autorisé.

PARTIE I : ALGORITHMIQUE (10 PTS)

Exercice 1. : (06 pts)

1. Définir : structure de données. 0,5 pt
2. Donner la principale différence entre un **File** et une **Pile**. 1 pt
3. Etablir une différence entre « **déclaration d'une procédure** » et « **appel d'une procédure** ». 1 pt

Votre professeur d'Histoire voudrait automatiser le traitement des notes de ses élèves de la classe. Ne connaissant pas l'effectif de la classe, votre camarade KISITO propose d'enregistrer les notes des élèves dans un tableau et saisir -1 pour terminer l'enregistrement.

- 3.1. Proposer une boucle adéquate pour parcourir ce tableau. 0,5 pt
- 3.2. Ecrire un sous-algorithme qui déclare un tableau de 1000 réels dont on fait ensuite saisir les valeurs par l'utilisateur. Les notes saisies sont comprises entre 0 et 20. 1,5 pt
NB : on arrêtera de remplir le tableau lorsque la valeur -1 sera saisie
- 3.3. En supposant les notes saisies, écrire un sous-algorithme qui calcule et retourne la moyenne générale de la classe 1,5 pt

Exercice 2. : (04 pts)

1. Ecrire un algorithme qui calcule et affiche la quantité de matière n_{al} d'un alcane dont le nombre d'atomes de carbone n et la masse m (en g) sont saisis par l'utilisateur. On rappelle qu'un alcane a pour formule générale C_nH_{2n+2} . Indication : la quantité de matière n_{al} d'un alcane est le rapport de la masse m par la masse molaire moléculaire M_{al} . 2 pts
C : 12 g/mol H : 1 g/mol ..
2. Soient les algorithmes ci-après :

Algorithme passage-valeur
Var M : entier ;
Procédure P1 (nombre : entier) ;
Debut
 Si nombre < 0 Alors
 nombre ← - nombre ;
 FinSi
 Ecrire (nombre) ;
FinProcédure

Debut
 Lire (M) ;
 P1 (M) ;
 Ecrire (M) ;
Fin.

Algorithme passage-adresse
Var M : entier ;
Procédure P1 (**Var** nombre : entier) ;
Debut
 Si nombre < 0 Alors
 nombre ← - nombre ;
 FinSi
 Ecrire (nombre) ;
FinProcédure

Debut
 Lire (M) ;
 P1 (M) ;
 Ecrire (M) ;
Fin.

Exécuter chacun des algorithmes dans un tableau pour la valeur $M = -9$

2 pts

PARTIE II : PROGRAMMATION EN LANGAGE C (10 PTS)

Exercice 1 : (05.5 pts)

1. Définir : Langage de Programmation 0,5 pt
2. En dehors du langage C, citer deux (02) autres langages de programmation. 1 pt
3. Lister deux (02) bibliothèques utilisées dans le langage C. 1 pt
4. Citer deux (02) fonctions C qui permettent la récupération des données saisies par l'utilisateur. 1 pt
5. Donner la signification de **IDE** et citer un exemple d'IDE utilisés en C. 1 pt
6. Identifier la dernière instruction d'un programme C dont la fonction principale se définit comme suit : *int main (void)*. Donner le rôle de cette instruction. 1 pt

Exercice 2 : (04.5 pts)

1. Donner le bout de code en langage C permettant de déclarer un tableau de 20 caractères. 0,5 pt
2. Soit le bout de code suivant

```
        if (n%2==0) {  
            printf("%d est pair",n);  
        } else {  
            printf("%d est impair",n);  
        }  
    }
```

Donner l'équivalent de ce code en utilisant l'opérateur conditionnel " ? ".

1 pt

3. Ecrire le programme C permettant de vérifier qu'un nombre saisi est Parfait ou non. Pour rappel, un nombre est parfait lorsqu'il est égal à la somme de ses diviseurs stricts. Exemple : 6 est parfait car $1+2+3 = 6$ 1,5 pt
4. Ecrire le code en langage C équivalent à l'algorithme ci-dessous :

Algorithme test

Var n, q, r : entiers ;

Début

Répéter

Ecrire ("Entrez un entier positif :") ;

Lire(n) ;

Jusqu'à (n >= 0)

Tantque (n >= 10) **faire**

q ← n div 10 ;

r ← n mod 10 ;

Ecrire(r) ;

n ← q ;

FinTantque

Ecrire(n) ;

Fin

1,5 pt