COLLEGE DE LA RETRAITE **MINI SESSION INTENSIVE N° 02 - ANNEE SCOLAIRE 2020-2021**

B.P. : 159 Yaoundé - Tél. : 243.58.86.54

EPREUVE D'ALGORITHMIQTE - Niveau PTI

Durée : 03H00

EPREUVE D'ALGORITHMIQUE

PARTIE 1 : GENERALITES SUR LES STRUCTURES DE DONNEES (8pts)

1. Dans le contexte informatique, définir structure de données, Tableau. (0.5*2=1pt)
2. Citer deux exemples de structures de données (0.25*2=0.5pt)
3. Donner la déclaration d'un tableau dont les caractéristiques sont les suivantes : (0.5pt)
   - ✓ Taille : 12 ;
   - ✓ les indices débutent à 3 ;
   - ✓ Le type de données est réel ;
4. On considère un tableau Tab de taille 10 dont les indices vont de 1 à 5. Indiquer une instruction permettant d'affecter la valeur 5 à la cellule d'indice 3. (0.5pt)
5. Identifier et recopier la bonne réponse pour cette assertion :
   L'écriture Vect[25]←2 signifie : (0.5pt)
   - Déclarer un tableau de taille 25 dont les indices débutent à 2 ;
   - Attribuer à la cellule 2 la valeur 25 ;
   - Attribuer à la cellule 25 la valeur 2 ;
6. Indiquer la boucle la mieux adaptée pour le parcours d'un tableau ? pourquoi ? (1pt)
7. On considère les déclarations de types Enregistrement suivantes :

Type Classe=Enregistrement Code :Chaine Effectif :Entier FinEnregistrement	Type Eleve=Enregistrement Nom :Chaine Age :Entier uneClasse :Classe FinEnregistrement

   a. Donner la déclaration d'une variable maclassede TypeClasse ainsi que la déclaration d'une variable unEleve de type Eleve ; (0.5pt)
   b. Donner deux instructions permettant d'affecter à la variable maclasse les valeurs PTI et 12(0.5pt)
   c. Donner trois instructions permettant d'affecter à la variable unEleve les valeurs de l'élèveDany, ayant comme 19 ans d'age, et inscrit en classe de PTI.(1pt)
   d. Donner une suite d'instructions permettant d'afficher les informations détaillées de la variable unEleve (1pt)
   e. Proposer la déclaration d'une structure de données permettant de mieux traiter les 12 élèves de la classe de PTI (1pt)

PARTIE 1 : LA RECHERCHE DICHOTOMIQUE (5pts)

On considère le tableau suivant :

Indice	1	2	3	4	5	6	7	8	9	10
éléments	25	12	14	48	13	30	4	23	34	44

Supposez que vous êtes le processeur, alors vous devez suivre et mettre en pratique les instructions qui vous seront données. Le but ici est d'atteindre la cellule qui contiendra l'élément 25 d'une part

Voici les instructions

Etape 1 : Ranger les éléments du tableau dans un ordre croissant ;

une fois les éléments rangés,

Etape 2 : trouver l'indice du milieu en faisant $(indice_{debut}+indice_{fin})/2$, et prendre la partie entière du

t.me/KamerHighSchool

Résultat.

Etape 3 : comparer l'élément contenu dans cette case du milieu (case d'indice trouvé) à 25.

- Si égalité, on s'arrête car on a trouvé
- Si élément inferieur à 25, effacer toutes les cellules à gauche de cette cellule du milieu
- Si élément supérieur à 25, effacer plutôt toutes les cellules à droite de cette cellule du Milieu

Etape 4 : Si on n'a pas trouvé, on recommence l'étape 2, mais en se servant du nouveau tableau Restant après éffaçage.

Question : Exécuter ces étapes en indiquant à chaque fois : (2pts)

- ✓ La valeur du milieu obtenue ;
- ✓ Le nouveau tableau obtenu après avoir effacer un coté

Ecriture de l'algorithme de la recherche dichotomique

Vous devez maintenant écrire l'algorithme de la recherche dichotomique et on supposera que le tableau T est déjà rangé. Pour cela, vous devez répondre aux questions ci-après :

1. Ecrire une instruction qui affecte à la variable milieu, l'indice du milieu d'un tableau de taille N ;(0.5pt)
2. Donner la syntaxe de la boucle Tantque qui s'arrête lorsque l'une des conditions ci-après est correcte :(1pt)
   - ✓ Lorsque T[milieu]<X =X ;
   - ✓ $Indice_{debut}$<$indice_{fin}$
3. Ecrire à l'intérieur de cette boucle tantque, le squelette de la structure alternative complète qui vérifie si un élément T[milieu]<X :(1pt)
   - ✓ Lorsque T[milieu]<X <X, la structure affecte milieu à indicedebut
   - ✓ Lorsque T[milieu]<X <X est faux, la structure affecte milieu à indicefin
4. Après la fermeture de tantque, ecrire une autre structure alternative qui affiche « trouvé » si A=X et « non trouvé sinon.(O.5pt)

PARTIE 3 : LES ALGORITHMES DE TRI (7pts)

On considère les 3 algorithmes de tri et 2 modes de fonctionnements ci-après :

```
Algorithme Tri1
 Variables:
          i , j, temp : Entiers
début
 pour ( i allant de 5 à 1) faire
  pour (j allant de 2 à i) faire
   si (Tab[ j-1 ] > Tab[ j ]) alors
     temp← Tab[ j-1 ] ;
     Tab[ j-1 ]← Tab[ j ] ;
     Tab[ j ] ← temp
   Finsi
  finpour
 finpour
Fin
```

```
Algorithme Tri2
 variables:
          i , j , v :Entiers
début
 pour (i allant de 1 à 5) faire
    v ←Tab[ i ]
    j ←i ;
     Tantque( j>1 ET Tab[ j-1 ] > v )faire
          Tab[ j ] ←Tab[ j-1 ];
            j←j-1;
     FinTantque ;
     Tab[ j ]← v
  finpour
Fin
```

```
Algorithme Tri3
 local:  m, i , j , n, temp :  Entiers
début
 pour i de 1 jusquà n-1faire
  pour j de i+1 jusquà n faire
   si Tab[ j ] < Tab[ m ] alors
     m ← j ;
     temp ← Tab[ m ] ;
     Tab[ m ] ← Tab[ i ] ;
     Tab[ i ] ← temp
     m ←i ;
  Fsi
 fpour
 fpour
Fin
```

Les modes de fonctionnement

Notice 1 :

Etape 1 : vérifier à partir du début, si l'élément de la cellule actuelle est inferieur à l'élément de la Cellule suivante :

- Si telle est le cas, passer à la cellule suivante ;
- Si tel n'est pas le cas, permuter ces deux éléments, ensuite effacer le 0 du brouillon et écrire 1, enfin passer à la cellule suivante.
- Effectuer ainsi ce travail jusqu'à la fin du tableau ;
- Verrouiller la dernière cellule ;

Etape 2 : Recommencer l'étape 1 pour les cellules non verrouillées restantes, jusqu'à ce que toute les cellules seront verrouillées

Notice2 :

Etape 1 : A partir du début, Relever la valeur d'une cellule quelque part

Etape 2 : Tant que les valeurs des cellules situées à sa gauche lui sont supérieures, les décaler d'un pas vers la droite

Etape 3 : Insérer la valeur relevée dans la cellule supposée restée vide ;

Etape 4 : passer à la cellule suivante et recommencer l'étape 1

Question :

1. Définir algorithme de TRI (0.5pt)
2. Indiquer deux exemples d'algorithmes de tri (0.5pt)

On considère le tableau suivant :

Indice	1	2	3	4	5
éléments	25	12	14	48	13

1. Mettre la notice du tri bulle en pratique pour ce tableau en représentant à chaque fois, le nouveau tableau obtenu à la fin de l'étape 1. (2pts)
2. Mettre la notice du tri insertion en pratique pour ce tableau en représentant à chaque fois, le nouveau tableau obtenu à la fin de l'étape 3. (2pts)
3. En considérant l'algorithme Tri1 et Tr3, indiquer ce que font les instructions ci-après :(1pt)

```
temp← Tab[ j-1 ] ;
Tab[ j-1 ]← Tab[ j ] ;
Tab[ j ] ← temp
```

```
temp ← Tab[ m ] ;
Tab[ m ] ← Tab[ i ] ;
Tab[ i ] ← temp
```

4. Associer chaque notice à son algorithme parmi les trois Tri (0.5pt)
5. Indiquer le nom de chaque algorithme de tri (0.5pt)