|  |  |  |
| --- | --- | --- |
| L.M.M.NABEUL | DEVOIR DE SYNTHESE N°1 | **Classe** : 4SI |
| **Durée** : **2** H |
| **Algorithmique et programmation** | **BEN ARBIA ADEL** |

Exercice 1(4 points)

Donner l’analyse d’un module **récursif** pour convertir **une chaine donnée** en un **nombre**.

La chaine est composée uniquement de chiffres.

***Remarque*** : L’utilisation de la procédure prédéfinie **VALEUR** est autorisée uniquement pour **convertir un caractère de la chaine** en un nombre.

***Exemple*** : Ch ="**124587**" donc le résultat est **124587**.

Exercice 2 (6 points)

On donne l’algorithme suivant de la procédure tri à bulles ordre croissant :

1. **DEFPROC** TRI\_B (N :Entier ; Var T :TAB)
2. **REPETER**

V 🡨 VRAI

**Pour** i **de** 1 **à** N-1 **Faire**

**SI** T[i] > T[i+1] **ALORS**

X 🡨 T[i]

T[i] 🡨 T[i+1]

T[i+1] 🡨 X

V 🡨 FAUX

**FIN SI**

**Fin Pour**

N 🡨 N - 1

**JUSQU'A** V ou (N=1)

1. **Fin** TRI\_B

**Travail demandé** :

Soit **T** un tableau contenant **N** entiers pairs et impairs, **P** est le nombre d’entiers pairs dans le tableau T.

1. Apporter des modifications à l’algorithme TRI\_B pour organiser le tableau T en mettant les entiers pairs au début de T et les entiers impairs à partir de la fin comme indiqué dans l’exemple suivant.

***Exemple***:

N=12 , P=6

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| T | i=1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
| **28** | **14** | 89 | 81 | **82** | **26** | 87 | 91 | 29 | **46** | **52** | 63 |

Organisation de T :

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| T | i=1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
| **28** | **14** | **82** | **26** | **46** | **52** | 63 | 29 | 91 | 87 | 81 | 89 |

1. **a**- Modifier maintenant l’algorithme TRI\_B pour effectuer le tri à bulles des **P** entiers pairs puis les **N-P** entiers impairs de T.

**b**- Donner les deux appels de la procédure TRI\_B.

Résultats des deux appels de TRI\_B :

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| T | i=1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
| **14** | **26** | **28** | **46** | **52** | **82** | 29 | 63 | 81 | 87 | 89 | 91 |

1. Modifier maintenant l’algorithme TRI\_B pour effectuer le **tri à bulles dans les deux sens**, ordre croissant, d’un tableau **R** contenant **k** entiers.

Exercice3 (10 points)

On considère une matrice **M** de **N** lignes x **N** colonnes (N est un entier impair compris entre 5 et 10).

On se propose d’écrire un programme pour :

* Remplir la matrice **M** par **des lettres majuscules au hasard** avec des consonnes et des voyelles dans chaque ligne.
* Remplir un fichier de type texte "**Mot.txt**" par **N** mots majuscules de **N** caractères (un mot par ligne).
* Chercher l’existence de chaque mot du fichier "**Mot.txt**" dans une **ligne, colonne ou diagonale** de la matrice **M**. Si le mot existe, on remplit alors un fichier de type texte "**Resultat.txt**" par le mot trouvé suivit par la ligne (i), la colonne(j) ou la diagonale de M.

**Exemple** : N=5

Matrice **M**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **F** | E | O | R | Q  Fichier : "Resultat.txt"  **FRERE** : Colonne 1 **ORDRE** : Ligne 3  **FIDEL** : Diagonal  Fichier : "Mot.txt"  **GRAND**  **FRERE**  **ORDRE**  **FIDEL**  **ELLES** |
| **R** | **I** | U | P | O |
| **E** | **O** | **D** | **R** | **R** |
| **R** | B | E | **E** | R |
| **E** | D | B | Y | **L** |

1. Analyser le problème en modules.
2. Analyser chaque module envisagé.