

NF01 - Automne 2009

Examen Final - 2 heures

Documents et calculatrices autorisés

Utilisez quatre copies séparées, une copie par problème

Problème n°1 (4 points)

Ecrivez un programme pour résoudre un système d'équations triangulaire tel que :

$$\begin{array}{rcl} a_{11} * x_1 + a_{12} * x_2 + \dots + a_{1n} * x_n & = & b_1 \\ a_{22} * x_2 + \dots + a_{2n} * x_n & = & b_2 \\ \dots & & \dots \\ a_{n-1\ n-1} * x_{n-1} + a_{n-1\ n} * x_n & = & b_{n-1} \\ a_{nn} * x_n & = & b_n \end{array}$$

avec a_{ij} , b_{ij} ($i=1..n$, $j=1..n$) coefficients réels. La taille maximale de la matrice sera 5x5.

Précisions utiles :

- Les inconnues de ce système sont les x_i , $i = 1..n$. Leur liste peut être représentée par un vecteur, c'est-à-dire un tableau à une dimension.
- L'ensemble des coefficients a_{ij} de ces inconnues peut être représenté par une matrice A de taille $n \times n$, c'est-à-dire un tableau à deux dimensions.
- De même, la liste des coefficients b_i du second membre peut être représentée par un vecteur B, c'est-à-dire un tableau à une dimension.
- Enfin on suppose que le système a une solution unique (i.e.) car $a_{ii} \neq 0 \ \forall i=1..n$.

Le programme devra :

- 1) Demander à l'utilisateur de saisir la taille de la matrice comprise entre 2 et 5 inclus [1 pt].
- 2) Demander à l'utilisateur de saisir les coefficients de la matrice triangulaire A puis les valeurs du second membre b [1 pt].
- 3) Résoudre le système et afficher le vecteur x résultat [2 pts].

Indication : on a $x_n = b_n / a_{nn}$, puis, $x_{n-1} = (b_{n-1} - a_{n-1\ n} * x_n) / a_{n-1\ n-1}$ etc...

A titre d'exemple la solution du système

$$\begin{array}{rcl} 2 * x_1 + 1 * x_2 + 0 * x_3 & = & 1 \\ 2 * x_2 + 1 * x_3 & = & 1 \\ 1 * x_3 & = & 1 \end{array}$$

est : $x_1=0.50 \ x_2=0.00 \ x_3=1.00$.

Problème n°2 (5 points)

Ecrivez un programme en Pascal qui permet la manipulation de chaînes de caractères entrées au clavier. L'utilisateur peut utiliser des lettres minuscules et/ou des chiffres. On rappelle qu'une chaîne de caractères peut être considérée comme un tableau de caractères et être manipulée comme tel.

Votre programme doit détecter et remplacer les chiffres par des lettres majuscules avec les correspondances suivantes : $0 \rightarrow A$, $1 \rightarrow B$, ..., $9 \rightarrow J$.

- 1) Proposer une fonction binaire booléenne qui lors de son appel reçoit un caractère et retourne la valeur vraie si le caractère est un chiffre et fausse sinon [2 pts].
- 2) Proposer une procédure à un argument de type chaîne de caractères, qui, en utilisant la fonction définie à la question précédente, retourne la chaîne de caractères modifiée [2 pts].
- 3) Donner le programme principal qui permet à l'utilisateur de saisir une chaîne de caractères, et qui, en utilisant les propositions des questions précédentes, affiche la chaîne modifiée [1 pt].

Problème n°3 (6 points)

On veut créer et traiter un fichier d'élèves pour chaque UV de l'UTC. Chaque élève y sera défini par les caractéristiques suivantes : Nom, Prénom, Date de naissance, Note au Médian, Note au Final, note de TP, Moyenne des trois notes précédentes (calculée automatiquement). On supposera que le nombre d'élèves dans chaque UV est inférieur ou égal à 300. Le nom du fichier sera le nom de l'UV concernée.

On souhaite écrire un programme PASCAL proposant à l'utilisateur d'effectuer différentes actions. Pour ce faire, vous devez :

- 1) Définir les structures de données [1 pt].
- 2) Pour une UV donnée, créer les procédures ou les fonctions correspondant aux actions suivantes :
 - a) Créer une version initiale du fichier de cette UV, en saisissant les données relatives à un ensemble d'étudiants [1 pt].
 - b) Ajouter un étudiant au fichier de cette UV [1 pt].
 - c) Afficher la liste des étudiants n'ayant pas la moyenne dans cette UV [1 pt].
 - d) Rechercher un étudiant de cette UV à partir de son nom de famille et de son prénom. On supposera que deux étudiants peuvent avoir le même nom, mais qu'ils ne peuvent pas avoir à la fois le même nom ET le même prénom [1 pt].
 - e) Supprimer un étudiant du fichier de cette UV : à ce propos, vous pourrez effectuer cette tâche soit par la manipulation directe de fichiers, soit en passant temporairement par un tableau constitué à partir du fichier initial [1 pt].

Problème n°4 (5 points)

Soit un tableau contenant une liste d'entiers ordonnés : les valeurs augmentent jusqu'au sommet (le plus grand entier du tableau), et puis décroissent jusqu'au dernier élément du tableau, comme dans l'exemple suivant :

6	8	9	12	16	17	19	11	9	7	4	3
---	---	---	----	----	----	----	----	---	---	---	---

On vous demande d'écrire une fonction récursive qui renvoie la valeur de ce sommet, c'est-à-dire la valeur maximale de ce tableau, ici 19.

Remarque : un tableau contient un seul sommet, et ne contient pas deux valeurs successives égales. La valeur du sommet peut être à une place quelconque.

Aide générale :

De manière générale, pour trouver une solution récursive à un problème donné, on peut essayer de décomposer le grand problème en plusieurs petits problèmes qui eux-mêmes pourront être résolus de la même manière.

Aide spécifique :

Pour trouver le maximum du tableau de façon récursive, l'idée est tout d'abord de le décomposer en deux parties, constituées après détermination d'une position médiane qu'on notera *mid*. De plus, sachant que les valeurs du tableau n'ont qu'un unique sommet, celui-ci ne peut donc se trouver que dans l'une de ces deux parties, à laquelle on pourra restreindre la recherche de ce sommet.

Début de simulation :

6	8	9	12	16	17	19	11	9	7	4	3
---	---	---	----	----	----	----	----	---	---	---	---

Puis :

6	8	9	12	16	17	19	11	9	7	4	3
---	---	---	----	----	----	----	----	---	---	---	---

Etc...

Questions :

- 1) Donner une condition qui va orienter la recherche vers l'une des deux parties [1 pt].
- 2) Donner un critère d'arrêt de la récursion [1 pt].
- 3) Ecrire une fonction récursive qui renvoie la valeur de ce sommet, c'est-à-dire la valeur maximale de ce tableau [3 pts].