Examen fin de module Algorithmique et programmation Durée : 2H30

Aucun document n'est autorisé

Exercice 1 8pts

On souhaite définir un type structuré de données noté **varchar_t** pour représenter les chaines de caractères de longueur variable. Une chaine de longueur variable sera représentée par une structure contenant un tableau dynamique de caractères noté txt, la taille maximale de la chaine notée **max_size** et sa taille effective notée **size**. Le tableau txt doit toujours se terminer par '\0'. On souhaite définir pour le type **varchar_t** un ensemble d'opérations permettant la manipulation de la chaine dont les signatures sont :

- varchar-t create_varchar(size_t max_size);
 Création de la chaine étant donnée sa taille maximale. La chaine est initialisée à l'état vide (cad size=0)
- void scan_varchar(varchar_t *ptr_varchar);
 lit une chaine à partir du clavier et place dans la chaine variable dont l'adresse est reçue par scan_varchar.
 Si le flux de caractère lu sur le clavier est plus long seuls les premiers caractères seront placés dans la chaine dans la mesure de sa capacité. La fonction doit lire et placer tout caractère sauf '\n' qui est réservé pour marquer la fin de la lecture au clavier.
- int find_word(const varchar_t *const ptr_varchar, const char *word, int n); retourne la position de la n^{ème} occurrence de word dans la chaine pointée par ptr_varchar. Retourne -1 si le mot n'est pas trouvé.
- void replace_word(varchar_t *const ptr_varchar, const char *word1,const char *word2, int n); qui remplace la nème occurrence du mot word1 dans la chaine pointée par ptr_varchar, lorsqu'elle existe, par le mot word2.

Ecrire un programme C qui permet et teste les opérations ci-dessus. (48,49)(53,54) (43->41)

Exercice 2 12pts

On souhaite élaborer un programme pour résoudre un système linéaire de la forme :

$$A x = y \tag{1}$$

Le système (1) peut être écrit aussi sous la forme :

$$\begin{cases} a_{11}x_1 + a_{11}x_2 + \dots + a_{1n}x_n = y_1 \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n = y_2 \\ \dots \\ a_{n1}x_1 + a_{n2}x_2 + \dots + a_{nn}x_n = y_n \end{cases}$$
(2)

Professeur M.QBADOU Page 1

- A est une matrice carrée réelle de taille nxn,
- x et y sont deux vecteurs réels de taille nx1.
- A, y sont les données du système (connus) et x est l'inconnu du système

Plusieurs méthodes numériques ont été développées parmi lesquelles on considère la **méthode du pivot de Gauss** qui repose sur deux étapes :

• La triangularisation du système (1) : transformations de l'équation (1) sous une forme équivalente en y appliquant une suite de combinaison linéaires des lignes de A et de y jusqu'à ce que le système devienne triangulaire supérieur sous la forme suivante :

$$\begin{cases}
a_{11}x_1 + a_{11}x_2 + \dots + a_{1n-1}x_{n-1} + a_{1n}x_n = y_1 \\
0 x_1 + a_{22}x_2 + \dots + a_{2n-1}x_{n-1} + a_{2n}x_n = y_2 \\
0 x_1 + \dots + 0x_{i-1} + a_{ii}x_i + \dots + a_{in}x_n = y_i \\
0 x_1 + 0 x_2 + \dots + 0 x_{n-1} + a_{nn}x_n = y_n
\end{cases}$$
(3)

En effet, si on note Ai, yi les ièmes lignes de A et de y respectivement et Aj et yj les jèmes lignes, alors si on substitue Aj et yj respectivement par : $A_i + \alpha A_j$ et $y_i + \alpha y_j$,

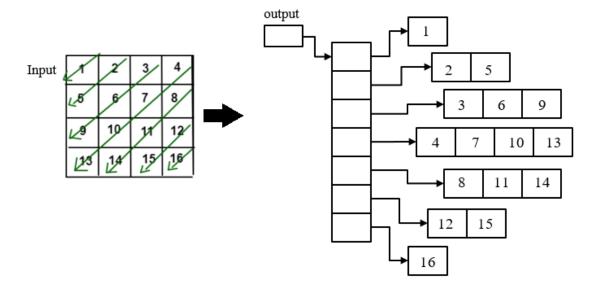
le système obtenu reste équivalent au système initial et ce pour tout i,j dans 1..n et pour tout réel α Si après triangularisation, les coefficients diagonaux sont non nuls, le système est dit de **Cramer** et possède une solution unique.

- Résolution du système (3) par substitution à partir de la fin (c'est-à-dire partant de $x_n, x_{n-1}, ... x_1$)
- 1. Analyser le problème et proposer une structure de données minimale qui représente un système linéaire de taille N quelconque (N variable)
- 2. Ecrire la fonction scan_sys qui permet de créer et scanner sur le clavier les données d'un système linéaire
- 3. Ecrire la fonction **print_sys** qui affiche le système à l'écran
- 4. Ecrire la fonction **find_max_pivot** qui cherche le long d'une colonne k à partir de la ligne k le numéro de la ligne de l'élément maximal.
- 5. Ecrire la fonction **invert_rows** qui intervertit deux lignes du système
- 6. Ecrire la fonction **combine_rows** qui recalcule une ligne Lj du système par la combinaison linéaire suivante : $L_j = L_j + \alpha L_i$
- 7. Ecrire la fonction **triangularize_sys** qui permet de calculer et retourner le système triangulaire équivalent à un système linéaire reçu en paramètre
- 8. Ecrire la fonction is_cramer qui vérifie si le système reçu en paramètre est un système de Cramer
- 9. Ecrire la fonction **solve_sys** qui résout le système linéaire reçu en paramètre.
- 10. Ecrire la fonction main qui teste les différentes fonctions.

Exercice 3 20pts

On considère une matrice carrée d'entiers de taille NxN. Ecrire une fonction qui retourne l'ensemble de ces vecteurs anti-diagonaux. Par exemple :

Professeur M.QBADOU Page 2



Professeur M.QBADOU Page 3