CONCOURS ESIM Entrepreneur Industrie - Session 2003

Option MP

EPREUVE D'INFORMATIQUE

Durée : 4 heures

L'usage des calculatrices est autorisé. Les algorithmes demandés peuvent être codés en CAML ou en PASCAL. Quand aucune directive n'est donnée le candidat peut établir une version récursive ou non. Quel que soit le choix du langage, les algorithmes doivent être écrits de la manière la plus courte possible, parfaitement lisible, avec une indentation convenable, sans aucune rature et en ***respectant scrupuleusement les notations introduites****. Ils doivent être documentés par des explications concises et précises sur les points qui le nécessitent.*
La lecture de l'annexe *correspondant au langage choisi est vivement conseillée avant d'aborder les questions de programmation.*
Les réponses qui ne respecteraient pas les consignes précédentes ne seront pas prises en considération.
Les parties I et II sont indépendantes.

I - Tranche de somme minimale.

Soit $V=(v_0, v_1, \cdots, v_n)$ un tableau dont les éléments sont des entiers relatifs indexés de 0 à n ($v_i \in \mathbb{Z}, 0 \le i \le n$). Une tranche T de V est une suite non vide d'éléments consécutifs de V, i.e. $T = (v_j, v_{j+1}, \cdots, v_k)$, avec $0 \le j \le k \le n$. On note $T=V(j..k)$ et E l'ensemble des tranches de V.

A toute tranche $T=V(j..k)$ de V on associe la somme $S(T)$ des éléments qui la composent : $S(T) = \sum_{i=j}^{k} v_i$.

Une tranche T de V est dite minimale si la somme qui lui est associée est minimale. Il peut ne pas y avoir unicité.
Le but de cette partie est de comparer divers algorithmes qui permettent de trouver une tranche minimale $T=V(d..f)=min\ \{ S(t),\ t \in E \}$.

1) Ecrire la fonction *somme* qui prend pour arguments un tableau V et deux nombres entiers j et k ($j \le k$) et qui calcule la somme associée à la tranche $V(j..k)$.

2) Ecrire, en utilisant la fonction *somme*, la fonction *tranche_min1* qui prend pour arguments un tableau V et un entier n (n désigne ici le nombre de composante du tableau V) et qui renvoie pour résultat la somme $S(T)$ associée à une tranche minimale T.

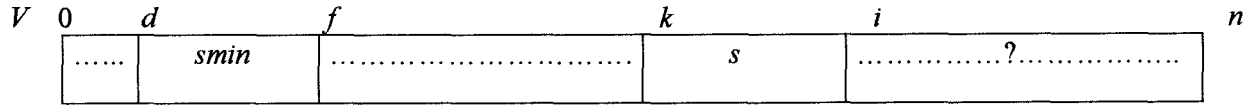
3) Déterminer en fonction de n, le nombre d'additions et de comparaisons effectuées suite à l'appel de la procédure *tranche_min1*.

4) Pour j fixé, $0 \le j \le n$. Ecrire la fonction *somme_min*, prenant pour arguments un tableau V et deux entiers j et n, qui sans utiliser la fonction *somme* écrite auparavant, calcule la somme d'une tranche minimale dont le premier élément est v_j, i.e. $\min_{j \le k \le n} S(V(j..k))$.

5) Ecrire, en utilisant la fonction *somme_min*, la fonction *tranche_min2* qui prend pour arguments un tableau V et un entier n (n désigne ici le nombre de composantes du tableau V) et qui renvoie pour résultat la somme $S(T)$ associée à une tranche minimale T.

6) Déterminer en fonction de n, le nombre d'additions et de comparaisons effectuées suite à l'appel de la procédure *tranche_min2*.

7) Soit le schéma :



Précisons la situation décrite par ce schéma. En supposant que toutes les composantes entre 1 et i ont été examinées, la tranche de somme minimale (en ne considérant que les éléments d'indices compris entre 0 et i) commence en d, se termine en f et a pour somme *smin*. Par ailleurs, parmi toutes les tranches qui s'achèvent en i, il en existe (au moins) une de somme minimale, on suppose qu'elle commence en k et a pour somme s.

a) Après incrémentation de i ($i \leftarrow i+1$), quelles sont les deux seules valeurs à envisager, pour déterminer parmi toutes les tranches qui s'achèvent en i, celle qui est de somme minimale.

b) En déduire un algorithme, en $O(n)$, permettant de trouver une tranche de somme minimale T du tableau V. Cet algorithme doit déterminer les indices d et f qui délimite la tranche minimale T, ainsi que *smin*=$S(T)$ la somme qui lui est associée.

c) Ecrire la fonction *tranche_min3* correspondant à cet algorithme.

II - Mots de Lyndon

Dans cet énoncé, un mot est une suite finie d'entiers $u = (u_1, u_2, \cdots, u_n)$ où $n \geq 1$ et $u_i \in \{0,1\}$ pour $i=1,\ldots, n$.
L'entier n désigne la longueur du mot u.
Si $u = (u_1, \cdots, u_n)$ et $v = (v_1, \cdots, v_m)$ sont des mots, on note uv le concaténé des mots u et v
$uv = (u_1, \cdots, u_n, v_1, \cdots, v_m)$.
Soient $u = (u_1, \cdots, u_n)$ et $v = (v_1, \cdots, v_m)$ deux mots.
Le mot u est un "début" de v si $n<m$ et $u_i = v_i$ pour $i=1,\ldots, n$.
Le mot u est une "fin" de v si $n<m$ et $u_i = v_{m-n+i}$ pour $i=1,\ldots, n$.
On pose $u \prec v$ *si* u est un début de v ou s'il existe un entier k tel que $u_i = v_i$ pour $i=1, \ldots, k-1$ et $u_k < v_k$.
La relation $\preceq$ définie par $u \preceq v$ si et seulement si $u=v$ ou $u \prec v$ s'appelle l'*ordre lexicographique*.
On pose $u<<v$ si et seulement si $u \prec v$, u n'étant pas un début de v. Ainsi, $u \prec v$ si u est un début de v ou si $u<<v$.
Un *conjugué* d'un mot $u = (u_1, \cdots, u_n)$ est un mot de la forme $v = (u_i, \cdots, u_n, u_1, \cdots, u_{i-1})$ avec $2 \leq i \leq n$.
Un mot est un mot de Lyndon si $u \prec v$ pour tout conjugué v de u.
Un mot de longueur 1 est évidement un mot de Lyndon.

Etude Mathématique :

1) Démontrer que pour deux mots $u \neq v$, on a soit $u \prec v$, soit $v \prec u$.

2) Soit u un mot. Démontrer que s'il existe un mot qui est à la fois un début et une fin de u, alors u n'est pas un mot de Lyndon.