IF121 : Informatique fondamentale

DEUG MIAS et MASS

Examen du 25 novembre 2000
durée 2 heures

Les documents sont autorisés.

Toutes les réponses devront être correctement justifiées. La rigueur des raisonnements, la clarté des explications, mais aussi la qualité de la présentation influeront sensiblement sur la note.

Le barème sur 21 n'est donné qu'à titre indicatif et pourra être modifié.

1 Conjugaison (3 points)

Écrire un programme en pseudo-langage qui lit un pronom (je, tu, il, elle, nous, vous, ils, elles) et le radical d'un verbe du deuxième groupe et affiche la forme conjuguée au présent de l'indicatif.

Il n'est pas si simple de savoir si une chaîne de caractères est le radical d'un verbe du deuxième groupe. Comme le temps est limité, on ne demande pas de le vérifier.

Par contre, le programme signalera une erreur si le pronom utilisé n'est pas dans la liste ci-dessus.

Voici quelques exemples d'exécutions du programme.

Données	Résultats
nous fin	nous finissons
je xyz	je xyzis
on garn	Ce n'est pas un sujet autorisé

2 Carré magique (6 points)

Une matrice carrée est un carré magique si ses lignes, ses colonnes et ses deux diagonales ont toutes la même somme appelée constante magique du carré.

Par exemple, la matrice suivante est un carré magique de constante magique 34 :

$$\begin{matrix} 16 & 3 & 2 & 13 \\ 5 & 10 & 11 & 8 \\ 9 & 6 & 7 & 12 \\ 4 & 15 & 14 & 1 \end{matrix}$$

Écrire en pseudo-langage un programme pour tester si une matrice de taille maximale 100 est un carré magique et, si c'est le cas, déterminer sa constante magique.

L'utilisateur devra fournir la taille ($<=$100) et la matrice ligne par ligne.

On écrira un programme complet avec toutes les déclarations et on détaillera la lecture de la matrice.

On ne demande pas de prouver que le programme est correct.

3 Invariant (4 points)

Voici un programme écrit en pseudo-langage.

```
constante Max = 50
type Tab = tableau de Max entiers
programme min2
variables
    t : Tab
    i,x,y : entiers
début min2
    lire t
    si t[1] < t[2] alors
        x  <-  t[1]
        y <- t[2]
    sinon
        x <- t[2]
        y <- t[1]
    finsi
    i <- 3
    tant que i <= Max faire
        si t[i] < x alors
            y <- x
            x <- t[i]
        sinon
            si t[i] < y alors y <- t[i] finsi
        finsi
        i <- i+1
    fin tant que
    Afficher x,y
fin min2
```

Montrer en utilisant un invariant que ce programme affiche les deux plus petits éléments du tableau t.

4 Programmation en C - - (2 points)

Traduire en C - - le programme de l'exercice 3.

5 Tri par insertion (6 points)

Écrire en pseudo-langage un programme qui lit un tableau, le trie par insertion et l'affiche. Le tri par insertion d'un tableau t procède par insertions successives des éléments $t[i]$ dans la partie déjà triée $t[1 \cdots i-1]$. Voici un exemple :

action	valeurs de t						
Initialement	7	2	5	12	6	15	1
insertion de $t[2]$ dans $t[1 \cdots 1]$	2	7	5	12	6	15	1
insertion de $t[3]$ dans $t[1 \cdots 2]$	2	5	7	12	6	15	1
insertion de $t[4]$ dans $t[1 \cdots 3]$	2	5	7	12	6	15	1
insertion de $t[5]$ dans $t[1 \cdots 4]$	2	5	6	7	12	15	1
insertion de $t[6]$ dans $t[1 \cdots 5]$	2	5	6	7	12	15	1
insertion de $t[7]$ dans $t[1 \cdots 6]$	1	2	5	6	7	12	15