



Licence
Sciences et Technologies

ANNEE : 2006/2007

SESSION DE JUIN 2007

ETAPE : INF2 MAT2 MAI4

UE : INF102

Epreuve : Initiation à l'algorithmique

Date : 29 Juin 2007

Heure : 11 H 00

Durée : 1 H 30

Documents : Tous documents interdits

Vous devez répondre directement sur le sujet qui comporte 4 pages.

Insérez ensuite votre réponse dans une copie d'examen comportant tous les renseignements administratifs

Epreuve de M^{me} Delest.

Indiquez votre code **d'anonymat** :

La notation tiendra compte de la clarté de l'écriture des réponses.

Barème

- Question 1 – Connaissances générales : 4 points
- Question 2 – Compréhension d'algorithme : 3 points
- Question 3 – Compréhension d'algorithme récursif : 4.5 points
- Question 4 – Ecriture de fonctions : 4 points
- Question 5 – Manipulation de listes : 4.5 Points

Question 1. Cochez les affirmations qui sont correctes :

- Le corps d'une boucle *tantque* est toujours exécuté une fois.
- L'*indice de boucle* peut-être une variable de type structure.
- Une variable passée par *référence* à une fonction est recopiée dans l'environnement de la fonction.
- La récursivité est un mécanisme qui peut conduire à une complexité en mémoire très pénalisante.
- La complexité minimale des algorithmes de tri vu en cours est $O(n)$
- La complexité minimale du tri rapide est $O(n)$
- Les listes simplement chaînées ont une complexité mémoire moins élevée que les listes doublement chaînées.
- Dans le tri à sélection, les k premiers éléments du tableau ne sont pas à leur place définitive après k itérations.

Question 2. Soit la fonction suivante :

```
fonction mystere(ref T:tableau[1..NMAX] d'entier, val clé :entier) : entier
    var d,f :entier ;
    début
        d:=1 ;f:=NMAX ;
        répéter
            tant que d<=NMAX et T[d]=<clé faire
                d=d+1 ;
            fintantque
            tantque f>=1 et T[f]>clé faire
                f=f-1 ;
            fintantque
            si d<f alors
                echange(T,d,f) ;
                d=d+1 ;
                f=f-1 ;
            finsi
        jusqu'à d>f ;
    retourner(f) ;
fin ;
finfonction
```

où la fonction *echange* est la fonction vue en cours qui échange deux éléments du tableau.

3. Réécrire cette fonction de manière itérative.

4. Comparer la complexité en mémoire de cette nouvelle fonction par rapport à la fonction récursive? Justifiez. Comparer la complexité en nombre de tests de cette nouvelle fonction par rapport à la fonction récursive? Justifiez.

Question 4.

Soit T un tableau de dimension N contenant des points en mouvement dans \mathbb{R}^3 décrits par leur coordonnées et leur vitesse exprimée par un nombre réel (on considère que les points se déplacent tous dans la même direction). Donnez une définition du type *pointEnMouvement* qui permet de décrire ces points. Ecrire une fonction *tropLoinTropVite* qui modifie le tableau T en supprimant les points qui sont à une distance supérieure à d de l'origine ou à une vitesse supérieure à v . Cette fonction retournera le nombre d'éléments ayant cette propriété.

Question 5.

1. Il est possible de gérer des listes simplement chaînées en utilisant un tableau de dimension fixe ? Justifiez et donnez les inconvénients

2. Soit L une liste doublement chaînée d'entiers.

- Que contient L et P après la séquence d'instructions suivante

```
var L : listeDC d'entier ;  
var P : ^entier ;  
creerListe(L) ;  
insereEnTete(7,L) ;  
insereEnTete(4,L) ;  
P=dernier(L) ;  
insereAvant (6,L,P) ;  
P=precedent(L,P) ;  
insereAprès(12,L,P) ;  
supprimerAprès(L,P) ;  
P=premier(L) ;  
supprimerAprès(L,P) ;  
P=suivant(L,P) ;
```

P=.....
Contenu(P)=
L=.....

3. Ecrire une fonction *puissanceDeux* dont le résultat est une liste contenant les puissances de deux dans l'ordre croissant et inférieures à un entier x .

4. Quel est le nombre d'éléments de la liste construite à la question 3 pour un entier x ?

.....

FIN