

EMP

Structures de données

Examen1 // 2009 2010

Barème probable : $(4 + 4+1) + (2 +6) + (2+1)$

Durée : 1H30

Tableaux

1. Créer un tableau ordonné à partir d'une suite de n nombres lus un à un.
2. La recherche 'trichotomique' dans un tableau ordonné consiste à diviser à chaque étape le tableau en trois intervalles 1 à $n/3$; $n/3 + 1$ à $2n/3$ et $2n/3 + 1$ à n . Si l'élément n'existe pas aux positions $n/3$ et $2n/3$ la recherche continue dans l'un des trois intervalles.
Donner l'algorithme itératif ou récursif.
Quel est le nombre maximum de comparaisons effectuées pour retrouver un élément ? Justifier

Listes

Donner les algorithmes permettant de :

1. Afficher les éléments d'une liste
2. Créer une liste linéaire chaînée ordonnée à partir d'une suite de n nombres lus un à un.

Liste de files d'attente de listes d'entiers

1. Comment implémenter en C une Liste de files d'attente de listes d'entiers ?
2. Donner la séquence d'opérations qui permet d'afficher le 3ième élément de la liste en tête de la troisième file d'attente.

Corrigé EMD 1 Algorithmique 2009 - 2010

SOIT

T UN VECTEUR (30) ;
N UN ENTIER ;
Inserer_tab UNE ACTION ;
Trichotomie , Trichotomie_rec DES ACTIONS ;
Afficher UNE ACTION ;
Inserer_liste UNE ACTION ;
Trouv UN BOOLEEN ;
L UNE LISTE ;

DEBUT

N := 30 ;
APPEL Inserer_tab ;
Ecrire (T) ;
APPEL Trichotomie (35 , Trouv) ;
Ecrire (Trouv) ;
APPEL Trichotomie_rec (35 , 1 , N , Trouv) ;
Ecrire (Trouv) ;
APPEL Inserer_liste (L) ;
APPEL Afficher (L) ;

FIN

{Afficher les éléments d'une liste}

ACTION Afficher (L)

SOIT

L UNE LISTE ;
P UN POINTEUR VERS UNE LISTE ;

DEBUT

P := L ;
TANTQUE P <> NIL
 Ecrire (VALEUR (P)) ;
 P := SUIVANT (P)
FINTANTQUE

FIN

ACTION Inserer_tab

SOIT

I , J , K , Nb DES ENTIERS ;
Val UN ENTIER ;
Trouv : BOOLEEN ;

DEBUT

Nb := 0 ;
POUR I := 1 , N
 Val := ALEANOMBRE (100) ;
 {Lire (Val) ;}
 J := 1 ;
 Trouv := FAUX ;
 TQ (J <= Nb) ET NON Trouv
 SI Val < ELEMENT (T [J])
 Trouv := VRAI
 SINON
 J := J + 1
 FSI
 FTQ ;
 SI NON Trouv
 AFF_ELEMENT (T [J] , Val) ;
 Nb := J
 SINON

```
POUR K := Nb, J, - 1
  AFF_ELEMENT ( T [ K + 1 ], ELEMENT ( T [ K ] ) );
```

```
FPOUR ;
AFF_ELEMENT ( T [ J ], Val );
Nb := Nb + 1
```

```
FSI
FPOUR
```

FIN

ACTION Trichotomie (Val , Trouv) ;

SOIT

```
Val UN ENTIER ;
Bi , Bs , Mil1 , Mil2 DES ENTIERS ;
Trouv UN BOOLEEN ;
```

DEBUT

```
Bi := 1 ;
Bs := N ;
Trouv := FAUX ;
TQ NON Trouv ET ( Bi <= Bs )
  Mil1 := ( 2 * Bi + Bs ) / 3 ;
  Mil2 := ( Bi + 2 * Bs ) / 3 ;
  SI ( ELEMENT ( T [ Mil1 ] ) = Val ) OU ( ELEMENT ( T [ Mil2 ] ) = Val )
    Trouv := VRAI
```

```
SINON
  SI Val < ELEMENT ( T [ Mil1 ] )
    Bs := Mil1 - 1
  SINON
    SI Val < ELEMENT ( T [ Mil2 ] )
      Bi := Mil1 + 1 ;
      Bs := Mil2 - 1 ;
```

```
SINON
  Bi := Mil2 + 1
```

```
FSI
FSI
FSI
FTQ
```

FIN

ACTION Trichotomie_rec (Val , Bi , Bs , Trouv) ;

SOIT

```
Val UN ENTIER ;
Bi , Bs , Mil1 , Mil2 DES ENTIERS ;
Trouv UN BOOLEEN ;
```

DEBUT

```
SI Bi > Bs
  Trouv := FAUX
SINON
  Mil1 := ( 2 * Bi + Bs ) / 3 ;
  Mil2 := ( Bi + 2 * Bs ) / 3 ;
  SI ( ELEMENT ( T [ Mil1 ] ) = Val ) OU ( ELEMENT ( T [ Mil2 ] ) = Val )
    Trouv := VRAI
```

```
SINON
  SI Val < ELEMENT ( T [ Mil1 ] )
    APPEL Trichotomie_rec ( Val , Bi , Mil1 - 1 , Trouv )
  SINON
    SI Val < ELEMENT ( T [ Mil2 ] )
      APPEL Trichotomie_rec ( Val , Mil1 + 1 , Mil2 - 1 , Trouv )
    SINON
      APPEL Trichotomie_rec ( Val , Mil2 + 1 , Bs , Trouv )
```

```
FSI
FSI
```

```

FSI
FSI
FIN
ACTION Insérer_liste ( L )
SOIT
L UNE LISTE ;
Q , P , Prec DES POINTEURS VERS DES LISTES ;
Trouv UN BOOLEEN ;
I UN ENTIER ;
Val UN ENTIER ;

```

```

DEBUT
L := NIL ;
POUR I := 1 , N
  Val := ALEANOMBRE ( 100 ) ;
  {LIRE ( Val ) ;}
  P := L ;
  Trouv := FAUX ;
  Prec := P ;
  TQ ( P <> NIL ) ET ( NON Trouv )
    SI Val >= VALEUR ( P )
      Prec := P ;
      P := SUIVANT ( P ) ;

```

```

  SINON
    Trouv := VRAI
  FSI ;

```

```

FTQ ;
ALLOUER ( Q ) ;
AFF_VAL ( Q , Val ) ;
AFF_ADR ( Q , P ) ;
SI Prec <> P
  AFF_ADR ( Prec , Q )
SINON
  L := Q
FSI ;

```

```

FPOUR ;

```

FIN

Pour la dernière question

Acces

F := Valeur(suivant(suivant(L))) // pour avoir la troisième file d'attente

Defiler(F, L2) // Pour avoir la liste en tête de la file d'attente

Ecrire(Valeur(Suivant(suivant(L2))) // pour afficher la troisième valeur de la liste