

Question de cours (04 points)

- 1- **Liste** : on ajoute et on supprime n'importe où(0.5)
Plie : ajoute à à fin, supprime à la fin (ou **LIFO**) ... (0.25 + 0.25)
File : ajoute à la fin, supprime au début (ou **FIFO**).. (0.25 + 0.25)
- 2- les éléments des enregistrements peuvent être de différents types, ceux du tableau sont du même type. (01)
- 3- **Fonction suite** (n : entier) : entier ; ... (0.25)
Debut
Si (n=0) **alors** ... (0.25)
 Suite ← 2 ; .. (0.5)
Sinon
 Suite ← suite (n-1) * 3 - 1 ; ... (0.5)
Finsi
 Fin.

Exercice 01 (06 points)

Algorithme exo2 ;

Var A : tableau [1..50,1..50] : entier ;(0.5)
 At : tableau [1..50,1..50] : entier;(0.5)
 i, j, n, m : entier(0.5)

Début

Ecrire (' donnez la taille de la matrice ');

Repete

 Lire (n); (0.25)

Jusqu'à (n<=50) et (n>0)(0.25)

Repete

 Lire (m); (0.25)

Jusqu'à (m<=50) et (m>0)(0.25)

 Pour i ← 1 à n faire
 pour j ← 1 à m faire
 lire (A [i, j]);
 fin pour
 finpour

 Pour i ← 1 à n faire (0.5)
 pour j ← 1 à m faire (0.5)
 At [j, i] ← A [i, j] ; ..(01) ou
 fin pour
 finpour

Pour i ← 1 à m faire
 pour j ← 1 à n faire
 At [i, j] ← A [j, i];
 fin pour

finpour

 Pour i ← 1 à n faire(0.5)
 pour j ← 1 à m faire(0.5)
 ecrire (At [i, j]);(0.50)
 fin pour
 finpour

Fin.

Algorithme exo3;

var tab : tableau [1..40] : entier ;(0.25)

m : entier; (0.25)

fonction existe (t : tableau [1..40] : entier ; n, x : entier): **booléen**;(0.5)

var i: entier ;(0.25)

debut

existe \leftarrow faux;(0.25)

pour i \leftarrow 1 à n **faire**(0.25)

Si (t[i] = x) **alors**(0.5)

existe \leftarrow vrai(0.25)

Finsi

Finpour

Fin_existe;

procedure lire (**var** t : tableau [1..40] : entier ; n: entier); (0.25)

var i, j, z : entier;(0.25)

Début

j \leftarrow 1;(0.25)

Tantque (j \leq n) **faire**(0.5)

Lire (z);(0.25)

Si (existe(t, n, z) = false) **alors**(0.5)

t [j] \leftarrow z;(0.25)

j \leftarrow j + 1 ;(0.25)

Finsi

Fin_tanque

Fin_lire;

procedure inverse (**var** t : tableau [1..40] : entier ; n: entier);(0.25)

var i, j, temp : entier ;(0.25)

Début

J \leftarrow n;(0.25)

pour I \leftarrow 1 à (n div 2) **faire**(0.5)

temp \leftarrow t[i];(0.25)

t[i] \leftarrow t[j];(0.25)

t[j] \leftarrow temp;(0.25)

j \leftarrow j - 1 ;(0.5)

finpour ;

fin_inverse ;

procedure **ecrire** (t : tableau [1..40] : entier; n : entier);(0.25)

var i: entier;(0.25)

Debut

pour I \leftarrow 1 à n **faire**(0.25)

ecrire (t[i]);(0.5)

finpour ;

fin_ecrire ;

debut

repete

lire (m);(0.25)

jusqu'à (m>0) et (m<=40);(0.25)

lire (tab , m) ;(0.25)

inverse (tab , m) ;(0.25)

ecrire (tab , m) ;(0.25)

Fin.