

Série N°4 TD Algorithme 1

Objectifs : Résolution de problèmes algorithme (Analyse), les structures séquentielles, les boucles, les tableaux à une et deux dimensions.

Rappel et Avertissement: La présence et la préparation des TD/TP est obligatoire. Celui ou Celle qui ne respecte pas les règles sur deux séances, ne fera pas d'évaluations ni intégrations d'algorithme.

NB : Bonus de + 5 à la note finale pour celui/celle qui trouve 5 sur les 9 exercices

Exercice 1 :

Écrire un algorithme permettant de saisir N notes, de calculer leur somme et leur moyenne de ces notes

Exercice 2 :

Écrire un algorithme pour chaque énoncé suivante :

- Affichage d'un rectangle de 'r' de largeur n et de hauteur p données.
- Affichage d'un triangle rectangle de 't' de largeur n et de hauteur p données.
- Affichage d'un carré de côté n/2 centré dans un carré de côté n+1.

Exercice 3 :

Écrire un algorithme qui remplace dans un tableau toutes les occurrences d'une valeur x par une valeur y.

Exercice 4 :

Écrire un algorithme **PURGER** qui ne conserve dans un tableau qu'un unique exemplaire de chaque valeur qu'il contient. La taille n du tableau sera modifiée en conséquence et la fin du tableau "abandonnée".

Exercice 5 :

Écrire un algorithme qui donne la position, dans un tableau T de taille n, de la première occurrence d'une valeur x donnée. Que fera l'algorithme si la valeur n'a pas été trouvée ?

Exercice 6 :

Écrire un algorithme qui supprime la k-ième valeur d'un tableau T de taille $n \geq k$ et déplace en conséquence les valeurs suivantes d'une case vers le début du tableau.

Exercice 7 :

algorithme Diagonale

Entrée : un tableau T de n valeurs.

Sortie : un tableau U de $n \times n$ valeurs tel que $U[i][i] = T[i]$.

Initialement : $U[i][j] = 0$ pour tous $i, j \in [0..(n-1)]$.

$i \leftarrow 0$;

Tantque ($i < n$) faire

```

j ← 0;
Tantque j(<n) faire
    Si (i=j) alors
        U[i][j] ← T[i];
    FinSi
    i ← i+1;
    j ← j+1;
FinTantque

```

FinTantque

- L'algorithme ci-dessus fait-il bien ce qu'il annonce ?
- Pourquoi les boucles Tantque ne sont-elles pas adaptées pour cet algorithme ?
- Écrire un algorithme équivalent qui remplace les Tantque par des structures de contrôle plus adaptées.

Exercice 8 :

Donnée : Deux entiers naturels n et p.

Résultat : x tel que... // T et F sont deux tableaux

```

i ← 0;
T[i] ← n;
F[i] ← p;
Tantque T[i]>1 faire
    T[i+1] ← T[i] div 2;
    F[i+1] ← F[i]+F[i];
    i ← i+1;

```

FinTantque

x ← 0;

Tantque i ≥ 0 faire

```

Si(T[i] mod 2 ≠ 0) alors
    x ← k+F[i];

```

FinSi

i ← i-1;

FinTantque

- Faire tourner l'algorithme avec n=35 et p=17.
- Que fait ce algorithme ? Comment le prouver ?
- On veut formaliser l'algorithme qui permet à un écolier sénégalais d'effectuer la multiplication de deux entiers n et p.
Comment modéliser n et p et le résultat n*p ?
Comment écrire l'algorithme ?

Exercice 9 :

On veut trier un tableau T de n entiers naturels non nuls en le copiant dans un autre tableau U tel que, pour chaque élément e de T, U[e]=e, et pour i n'appartenant pas à T, U[i]=0.

- Comment déterminer la taille de U ?
- Écrire un algorithme qui effectue ce transfert.
- Écrire un algorithme qui copie les n valeurs utiles de U dans un tableau W de taille n.
- Montrer que W réalise un tri par ordre croissant des valeurs de T.