

ESIEA

ALGORITHMIQUE ET PROGRAMMATION-INF2033

TD1

EXERCICE 1

- 1) Ecrivez dans le langage C une **fonction récursive** de nom **OCCU** qui retourne le **nombre de fois où X apparaît dans le tableau T[1 ... n]**.
- 2) Complétez la fonction main.

EXERCICE 2

Soit T[1 ... n] un tableau non ordonné de n réels, et X un réel donné.

- 1) Ecrivez dans le langage C une **fonction récursive** de nom **NBM** qui retourne le **nombre de majuscules dans une chaîne de caractères**.
- 2) Complétez la fonction main.

EXERCICE 3

Un mot est dit palindrome si la suite de ses lettres, obtenu par lecture de gauche à droite, est la même que par lecture de droite à gauche. Par exemple, le mot RADAR est un mot palindrome.

- 1) Ecrivez une **fonction booléenne récursive** de nom **PALINDROME** qui permette de **savoir si un mot donné est un mot palindrome**.
- 2) Complétez la fonction main.

EXERCICE 4

Soit T[1 ... n] un tableau ordonné de n entiers, et X un entier donné.

- 1) Ecrivez dans le langage C une **fonction récursive** de nom **DICHREC** qui **recherche par dichotomie** l'élément X dans le tableau T[1 ... n] dont les éléments sont triés en ordre croissant. **DICHREC doit délivrer la valeur de la place de**

l'élément X dans le tableau T[l ... n] ; C'est -1 si $X \notin T[l \dots n]$, et c'est $i \in \{l, \dots, n\}$ si $X \in T[l \dots n]$.

- 2) Complétez la fonction main.
- 3) Reprendre l'exercice face à un **tableau dynamique de dimension 2** qui est **trié dans l'ordre croissant**.

EXERCICE 5

La multiplication Egyptienne de deux entiers naturels a et b est fondée sur les relations suivantes :

- ❖ Si $b = 0$, alors $a * b = 0$
- ❖ Si b est impair, alors $a * b = a * (b - 1) + a$
- ❖ Si b est pair, alors $a * b = 2 * a * (b / 2)$

- 1) Ecrivez dans le langage C la **Fonction Récursive** correspondante.

EXERCICE 6

Le but de cet exercice est de résoudre le problème des tours de Hanoï en utilisant un algorithme récursif. Rappelons d'abord le principe de ce jeu.

N disques de diamètres deux à deux différents sont initialement enfilés sur la « tige de gauche ».

Ils sont disposés de telle sorte que leurs diamètres décroissent de la base de la tige vers le sommet de la tige (voir figure 1).

Le but du jeu est de déplacer ces N disques vers la « tige de droite » en se servant de la « tige du milieu », et en respectant les règles suivantes :

- ❖ Seul un disque peut être déplacé à la fois ;
- ❖ Un disque ne doit jamais être placé sur un disque de diamètre plus petit ;
- ❖ Chaque disque doit à tout moment se trouver sur une tige.

- 1) Simulez le problème pour 2 disques, puis pour 3.
- 2) Eu égard à la simulation, essayez d'analyser le problème.
- 3) En utilisant la fonction **DEPLACER(X, Y)** qui permet de déplacer un disque de la tige X vers la tige Y, écrivez dans le langage C une **fonction récursif HANOI(N, A, B, C)** qui déplace N disques de la « tige de gauche » vers la « tige de droite » en se servant de la « tige du milieu. On notera que la fonction HANOI prend en entrée quatre arguments :

- ❖ N qui représente le nombre de disques à déplacer et qui est de type entier ;
- ❖ A qui représente la « tige de gauche » et qui est de type caractère ;
- ❖ B qui représente la « tige du milieu » et qui est de type caractère ;
- ❖ C qui représente la « tige de droite » et qui est de type caractère.

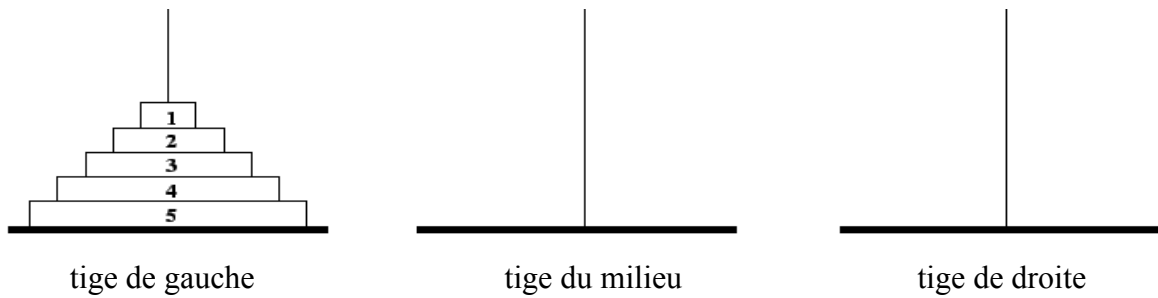


Figure 1. Tours de Hanoi. Les disques sont numérotés de 1 à N (ici, $N = 5$).

4) Complétez la fonction main.