

## TD5 - Tableaux – Tri rapide

### 1 Quelques algorithmes de tableaux

EXERCICE 1 *Écrire un algorithme qui détermine tous les entiers premiers inférieurs à  $N$  (fixé), en utilisant un tableau. (méthode du Crible d'Eratosthène). Ensuite, afficher ces entiers premiers.*

Dans la suite les mots seront supposés codés sous forme d'un d'un tableau de caractères de taille  $N$  (fixé, grand) dont le dernier est égal à `'\0'` (caractère spécial de fin de chaîne).

EXERCICE 2 *Écrire un algorithme pour calculer la taille d'un mot.*

EXERCICE 3 *Écrire un algorithme pour renverser un mot en place. Le mot à inverser et la taille de ce mot seront passés en paramètres.*

EXERCICE 4 *Écrire un algorithme pour dire si un mot donné est un palindrome. On passera la taille de ce mot en paramètre.*

### 2 Tri rapide

Cette méthode consiste à placer un élément (appelé pivot) de la suite à trier à sa place définitive en permutant tous les éléments de telle sorte que tous ceux qui lui sont inférieurs soient à sa gauche et que tous ceux qui lui sont supérieurs soient à sa droite. Cette opération s'appelle partitionnement.

Pour chacune des sous-suites obtenues, on définit un nouveau pivot et on répète l'opération de partitionnement. Ce processus est répété récursivement, jusqu'à ce que la suite soit triée. En pratique, le pivot est soit le premier, soit le dernier élément de la suite à trier. Dans l'exemple suivant, le pivot est le dernier élément de la suite.

	<i>1</i>	<i>2</i>	<i>3</i>	<i>4</i>	<i>5</i>	<i>6</i>	<i>7</i>	<i>8</i>	
Tri rapide	28	13	20	35	21	19	15	18	<i>pivot</i> = 18
Partition	15	13	18	20	35	21	19	28	<i>pivot placé</i>
Tri rapide	15	13							<i>pivot</i> = 13
Partition	13	15							<i>pivot placé</i>
Tri rapide				20	35	21	19	28	<i>pivot</i> = 28
Partition				20	19	21	28	35	<i>pivot placé</i>
Tri rapide				20	19	21			<i>pivot</i> = 21
Partition				20	19	21			<i>pivot placé</i>
Tri rapide				20	19				<i>pivot</i> = 19
Partition				19	20				<i>pivot placé</i>
Fin	13	15	18	19	20	21	28	35	