

## TDP N° 5

### Traitements des chaînes de caractères

#### Exercice 0 :

- Donnez un automate qui reconnaît les mots qui sont la représentation binaire d'un nombre pair.
- Donnez un automate qui reconnaît les mots qui sont la représentation binaire d'un nombre multiple de 3.

#### Exercice 1 :

- Construire l'automate de recherche du motif  $P = aabab$  sachant que l'alphabet est :  $\{a, b\}$
- Illustrer son action sur le texte  $T = aaababaabaababab$

chaîne	état
$\epsilon$	
a	
aa	
aaa	
aaab	
...	

- En conclure le nombre d'apparition du motif  $P$  dans  $T$

#### Exercice 2 :

1. Ecrire les fonctions vues au cours :

- o Suffixe : qui vérifie si une chaîne  $chp$  est préfixe d'une chaîne  $CH$
- o Fonction Transition : `CALCUL-FONCTION-TRANSITION()`
- o `RECHERCHE-AUTOMATE-FINI()` : cette fonction doit **préciser le nombre de fois** qu'apparaît le Motif (pattern) dans le texte.

2. Ecrire un programme qui :

- o demande la longueur du motif, la longueur du texte, le motif (sur l'alphabet  $\{a, b, c\}$ ) puis
- o génère un texte de façon aléatoire,
- o affiche les cent premiers caractères de ce texte puis,
- o indique le nombre d'occurrences du motif dans le texte, pour la méthode naïve d'abord et pour la méthode avec automate ensuite,
- o compare le temps d'exécution entre les deux méthodes

**Exercice 3 :**

- Ecrire une fonction qui calcule le nombre de Fibonacci vu au cours, en utilisant la technique de memoïsation, puis compare le temps d'exécution avec la méthode récursive naïve
- Réécrire les 4 algorithmes ( de Algo1 à Algo4 vus au cours) relatifs au Problème de Rendu de Monnaie, de sorte à récupérer et le nombre minimal des pièces et la manière de construire cette solution optimale,

**Exercice 4 :**

- Etablir la matrice de calcul de distance de **Levenshtein** entre les deux chaînes :
  - o "MOONDAY" et "SATURYAY"
  - o "ALGORITHM" et "LOGARITHM"
- Réécrire la solution 2 du calcul de la distance de **Levenshtein** vue au cours sans utiliser une matrice  $(n+1) \times (m+1)$
- Comparer le temps d'exécution avec la solution 1, proposée au cours pour des chaînes suffisamment longues.

**Exercice 5 :**

Soit la séquence suivante de longueur 25 nommée TXT : "TAGGGATAGGGACCTAGGGGTAXTA"

- Construire l'arbre de Huffman relatif à TXT
- Donner la représentation binaire de chaque caractère
- En déduire le gain de la représentation de la séquence TXT par le codage de Huffman comparée à celle par le codage ASCII (8 bits / lettre).