

Inscrivez **lisiblement** vos NOM et Prénom en tête de vos copies.

Exercice 1 : (Pattern matching – [4 points])

1. [1 point] Construire et dessiner l'automate de recherche de toutes les occurrences du motif **actttac**.
2. [3 points] Supposons maintenant que l'on recherche un motif parmi une famille de motif : $\{\underline{a}ctttac, actttcc, acttttc, aggtttac, aggtttcc, aggttttc\}$. En remarquant la similitude entre les différents motifs de la famille, proposez un automate déterministe qui permette de rechercher toutes les occurrences de ces motifs. *Indication* : tous les mots partagent la 1^{ère} lettre, les trois **t** d'affilée et la dernière lettre.

Exercice 2 : (Problème "Le chemin le plus long dans un graphe" [16 points])

Soit $G = (V, E)$ un graphe orienté avec $V = \{v_1, \dots, v_n\}$. On dit que G est ordonné si il vérifie les deux propriétés suivantes :

- chaque arc de ce graphe est de la forme $(i \rightarrow j)$ avec $i < j$
- tous les sommets sauf le sommet v_n ont au moins un arc sortant.

Ici, par souci de simplification, nous supposons qu'il existe un chemin allant de v_i vers v_n pour tout $i = 1, \dots, n$.

L'objectif est de trouver le chemin le plus long entre les sommets v_1 et v_n .

1. [1 point] Montrer que l'algorithme glouton suivant ne résout pas correctement le problème.
Indication : un exemple où l'algorithme suivant ne donne pas le bon résultat peut être suffisant si on explique sur cet exemple le déroulement de l'algorithme.
 - (a) $u \leftarrow v_1$;
 - (b) $L \leftarrow 0$;
 - (c) Tant qu'il existe un arc sortant du sommet u
 - i. choisir l'arc $(u \rightarrow v_j)$ tel que j est le plus petit possible
 - ii. $u \leftarrow v_j$;
 - iii. $L \leftarrow L + 1$;
 - (d) retourner L
2. [1 point] Donner la formule de récurrence qui permet de calculer la longueur du chemin le plus long commençant par v_1 finissant par v_l .
3. [3 points] Donner un algorithme qui retourne la longueur du chemin le plus long commençant par v_1 finissant par v_n . En donner la complexité.
4. [2 points] Modifier l'algorithme précédent afin qu'il retourne le chemin. En donner la complexité.

Nous allons considérer les graphes orientés et ordonnés possédant une fonction de poids w sur les arcs $w : E \rightarrow \mathbb{N}^+$. L'objectif est de trouver le poids du chemin de poids maximal commençant par v_1 finissant par v_n (s'il n'en existe pas, la valeur retournée doit être égale à $-\infty$). Formellement, on veut trouver $p_{max} = \max\{\sum_{e \in \mathcal{P}} w(e) \mid \mathcal{P} \text{ est un chemin de } v_1 \text{ à } v_n\}$

5. [1 point] Donner la formule de récurrence permettant de calculer le chemin de poids maximum commençant par v_1 finissant par v_n .
6. [3 point] En déduire l'algorithme et donner sa complexité (argumentez).

Maintenant, l'objectif est de trouver le poids du chemin de poids maximal commençant par v_1 composé de k arcs. Nous utiliserons un tableau T dont la valeur de l'élément $T[i, l]$ correspond au poids du chemin de poids maximal commençant par v_1 finissant par v_i et composé de l arcs.

7. [2 points] Donner la formule de récurrence qui permet de calculer le chemin de poids maximal de l arcs commençant par v_1 finissant par v_i .
8. [3 points] En déduire l'algorithme.