

## L2 Informatique - Algorithmique - Session 2

jeudi 30 juin 2016. 13h30-15h30. X 300.

La précision et la clarté de votre rédaction sont *fondamentales* et vous devez *justifier* vos réponses. Documents interdits. Durée 2h00. Le barème indiqué est *approximatif*.

**Exercice 1.** [3pts] Soit  $q$  un nombre réel. Calculez la somme des  $n$  premières puissances de  $q$ , i.e. pour  $i \in [0, n-1]$ , que vaut cette somme si  $q = 1$ ? Soit  $n$  un entier naturel. Combien existe-t-il de séquences binaires de poids de Hamming  $k$ ,  $0 \leq k \leq n$ ? Justifiez vos réponses.

**Exercice 2.** [5pts] On dispose d'une liste  $L$  *croissante* de  $n$  valeurs réelles  $a_1, a_2, \dots, a_n$  strictement positives. Un *indice barycentrique* de la liste  $L$  est un entier  $k$  qui minimise la quantité

$$(1) \quad \left| \sum_{i=k+1}^n a_i - \sum_{i=1}^k a_i \right|.$$

L'entier  $k$  est-il unique? Écrivez un algorithme `Barycentre(L)` qui renvoie un indice barycentrique. Quelle est la complexité de votre algorithme? Utilisez `Occurrences(L, x)` pour écrire un algorithme `Majoritaire(L)` qui renvoie l'élément majoritaire de la liste  $L$  s'il existe et 0 sinon. On supposera donc que la liste ne contient pas la valeur nulle. Quelle est la complexité de votre algorithme (justifiez)?

**Exercice 3.** [2pts] Transformez le tableau  $T = [3, 1, 6, 2, 4, 5, 7]$  en tas en décomposant votre travail. L'indexation du tableau commence à 1. Quel est l'indice de la première valeur à laquelle il faut appliquer l'algorithme `Tamiser`?

**Exercice 4.** [4pts] Soient  $u = u_1u_2\dots u_n$  et  $v = v_1v_2\dots v_m$  deux mots sur un alphabet  $A$ . On dit que  $u$  est une *portion* de  $v$ , s'il existe un entier  $k$  tel que  $1 \leq k \leq m - n + 1$  et  $\forall i \in [1, n], u_i = v_{k+i-1}$ . Combien y-a-t-il de portions de longueur  $n$  d'un mot de longueur  $m$ ? Écrivez un algorithme `EstPortion(u, v)` qui renvoie `vrai` si  $u$  est une portion de  $v$  et `faux` sinon. On notera  $u[i]$  le  $i$ -ème terme de la séquence  $u$ . Estimez la complexité de cet algorithme dans le meilleur des cas et dans le pire des cas en fonction des longueurs  $n$  et  $m$  des mots  $u$  et  $v$ .

**Exercice 5.** [6pts] On suppose qu'une liste  $S$  contient une permutation des entiers  $\{1, 2, \dots, n\}$  et que l'indexation de la liste commence à 1. Soit  $k$  un entier,  $k \geq 2$ . On appelle *k-cycle* (ou cycle de longueur  $k$ ) de la liste  $S$ , toute liste  $[x_0, x_1, \dots, x_{k-1}]$  d'entiers tous distincts dans  $\{1, 2, \dots, n\}$  tels que

$$\forall i \in [0, k-1], S[x_i] = x_{(i+1) \bmod k}.$$

Par exemple, la liste  $[1, 8, 2]$  est un 3-cycle de la liste

$$S = [8, 1, 9, 7, 3, 6, 5, 2, 4].$$

Calculez l'autre cycle de cette liste.

Soit  $S$  une liste permutation de longueur  $n$ . Quelle est la longueur maximale d'un cycle? Quelle liste ne contient aucun cycle?

Écrivez un algorithme `Cycles(S)` qui renvoie la liste de tous les cycles de la liste  $S$ . Avant d'écrire votre algorithme, expliquez en français la méthode que vous allez employer. Quelle est la complexité de votre algorithme (justifiez)?

**Rappel important!** pour toutes les questions de complexité, précisez :

- (1) ce que désigne la/les variable(s) de la fonction de complexité  $T$ ;
- (2) quelles instructions vous comptabilisez et pourquoi vous les considérez représentatives de l'algorithme;
- (3) s'il y a lieu de distinguer différentes complexités, meilleur des cas, pire des cas (le cas échéant, le complexité moyenne n'est pas demandée).