Examen Algorithmique avancée – Rattrapage
UPMC — Master d'Informatique —
Mai 2014 – durée 2h

Les seuls documents autorisés sont les polys de cours et la feuille manuscrite.
Le barême est donné à titre indicatif.

1 Questions diverses [6 points]

Question 1. Etant donné une file binomiale contenant n éléments, proposer une méthode qui retourne la liste triée des éléments. Donner la complexité de cette méthode dans le pire des cas.

Question 2. Etant donné un arbre bicolore contenant n éléments, proposer une méthode qui retourne la liste triée des éléments. Donner la complexité de cette méthode dans le pire des cas.

Question 3. Dans le processus de rééquilibrage d'un arbre (AVL ou bicolore), on utilise des opérations de *rotation*. Montrer que les opérations RG et RGD

1. conservent l'ordre infixe (symétrique)
2. diminuent de 1 la hauteur de l'arbre à la racine duquel est effectuée la rotation.

Question 4. On considère un alphabet de 2^p symboles et un texte T sur cet alphabet. Si chaque symbole apparaît avec la même fréquence dans T, quelle est la hauteur de l'arbre de Huffman codant T.

2 Dichotomie dynamique [8 points]

La recherche binaire (dichotomique) dans un tableau trié consomme un temps logarithmique, mais le temps d'insertion d'un nouvel élément est linéaire par rapport à la taille du tableau. On peut améliorer le temps d'insertion en conservant séparément plusieurs tableaux triés.

Plus précisément, on suppose que l'on souhaite implanter les opérations de recherche et d'insertion sur un ensemble E ayant n éléments. Supposons que n s'écrive $b_{k-1} \dots b_1 b_0$ en base 2 (*i.e* $n = b_{k-1}2^{k-1} + \dots + b_1 2^1 + b_0 2^0$). On a k tableaux triés $A_0, A_1, \dots, A_{k-1}$, de tailles respectives $2^0, 2^1, \dots, 2^{k-1}$. Chaque tableau A_i est soit plein, soit vide selon que $b_i = 1$ ou $b_i = 0$. Le nombre total d'éléments contenus dans les k tableaux est donc $b_0 2^0 + b_1 2^1 + \dots + b_{k-1} 2^{k-1} = n$. Bien que chaque tableau soit trié, il n'existe pas de relation particulière entre les éléments des différents tableaux.

Un exemple Si $E = \{1, 2, \dots, 13\}$ alors n s'écrit 1101 en base 2. Le tableau A_1 est vide et les tableaux A_0, A_2 et A_3 sont pleins, par exemple : $A_0 = [13]$, $A_2 = [1, 6, 8, 11]$, $A_3 = [2, 3, 4, 5, 7, 9, 10, 12]$ (mais ce n'est pas la seule manière de remplir A_0, A_2 et A_3).

Question 1. Recherche

1. Décrire une méthode de recherche opérant sur la structure de données présentée dans l'introduction.
2. Calculer la complexité de l'opération de recherche dans le pire cas.

Question 2. Insertion

1. On suppose que la complexité de l'interclassement de deux tableaux triés de tailles t_1 et t_2 est égale à $t_1 + t_2$. On suppose que les tableaux $A_0, A_1, \ldots, A_{i-1}$, de tailles $2^0, 2^1, \ldots, 2^{i-1}$, sont pleins et triés. Montrer que l'on peut réaliser leur interclassement en temps inférieur ou égal à 2^{i+1}. Comment faut-il procéder ?
2. (a) Décrire une méthode d'insertion d'un nouvel élément (distinct de tous éléments déjà présents) opérant sur la structure de données présentée dans l'introduction.

   (b) On suppose que A_0, A_1, A_2 et A_3 sont vides et que $A_4 = [1, 2, \ldots 16]$. Insérer 20.

   (c) On suppose que A_3 est vide et que $A_0 = [20]$, $A_1 = [3, 13]$, $A_2 = [1, 6, 8, 11]$ et $A_4 = [21, 22, \ldots 36]$. Insérer 12.
3. Calculer la complexité de l'insertion dans le pire cas :
   – en fonction de i, où i est tel que $A_0, A_1, \ldots, A_{i-1}$ sont pleins et A_i est vide
   – en fonction de n.

Question 3. Coût amorti

1. Une séquence d'opérations est effectuée sur une structure de données. Soit n un entier naturel et $b_{k-1} \ldots b_1 b_0$ l'écriture de n en base 2. Soit $\alpha(n)$ le plus petit indice i tel que $b_i = 0$ (si tous les $b_{k-1}, \ldots, b_1, b_0$ sont égaux à 1 alors $\alpha(n) = k$). On suppose que la n-ième opération coûte $2^{\alpha(n)}$.

   Montrer que, pour $n = 2^k - 1$, le coût total d'une suite de n opérations est égal à $k2^{k-1}$. En déduire le coût amorti d'une opération, lorsqu'on effectue une suite de n opérations.
2. On considère une suite de n insertions dans la structure de données présentée dans l'introduction (tous les tableaux étant initialement vides). Montrer que le coût amorti d'une insertion est en $O(\log n)$.

3 Couches convexes [6 points]

L'ensemble des sommets de l'enveloppe convexe d'un ensemble S est notée $EC(S)$.
Étant donné un ensemble fini S de points du plan, on définit les *couches convexes* de S par récurrence :
– on pose $S_1 = S$ et $C_1 = EC(S_1)$
– pour $i > 1$, on pose $S_i = S_{i-1} \setminus C_{i-1}$ et $C_i = EC(S_i)$ (si S_i est vide alors C_i est vide).

Pour $i \geq 1$, l'ensemble C_i est appelé *i-ème couche convexe* de S.

1. Dessiner les couches convexes d'un ensemble comportant une quinzaine de points (vous choisirez les points de telle sorte qu'il y ait plusieurs couches).
2. Montrer qu'il existe un entier i_0 tel que $C_i = \emptyset$ pour tout $i > i_0$.
3. Donner un algorithme, basé sur l'algorithme de Jarvis, permettant de calculer les couches convexes non vides d'un ensemble fini S. Quelle est sa complexité ?
4. Donner un algorithme, basé sur l'algorithme de Graham, permettant de calculer les couches convexes non vides d'un ensemble fini S. Quelle est sa complexité ?
5. Montrer que la complexité de tout algorithme procédant par comparaisons pour calculer les couches convexes d'un ensemble de n points est en $\Omega(n \log n)$.