UNIVERSITÉ DE MONTRÉAL
Département IRO

Été 2016
Neil Stewart

IFT 2015 E16

Devoir 1.

10/10, soit 10% de la note finale.

Les 10 points "Partie pratique" pour le cours E16 seront probablement distribués de la façon suivante : Devoir #1 (1 point), Devoir #2 (3 points), Devoir #3 (6 points).

1 Partie Pratique (1 point)

Dans ce devoir, nous allons trouver la valeur maximale dans une liste simplement chainée. Vous devez utiliser le squelette mis à votre disposition.

NOTE : Ne modifiez pas le squelette et travaillez *seulement* sur la classe *MyMainClass.java*.

Tâche :

1. Implanter la fonction *findMax(Element list)*.

Vous pouvez tester votre code en utilisant le fichier data.in disponible sur le site *StudiUM*.

Modalités de remise : envoyer une archive <nom1-prenom1_nom2-prenom2.zip> à l'adresse teodora.dan@umontreal.ca.

2 Partie Théorique (9 points)

1. (1 1/2 points)

   Le but de cet exercice est de souligner que les constants c de $O(\ldots)$ et $\Omega(\ldots)$ qui sont cachées dans $\Theta(\ldots)$ ne sont pas obligatoirement égales.

   Supposons qu'un algorithme doit traiter N items, et chaque traitement i, $1 \leq i \leq N$, exige $r[i]$ unités de temps, où cette quantité r varie de façon aléatoire dans l'intervalle $[1, 3]$. À chaque étape $r[i]$ peut prendre n'importe quelle valeur dans l'intervalle.

   (a) Démontrez que $T_{worst}(N)$ est $\Theta(N)$. Entre autre, précisez les valeurs des constantes c qui font l'affaire.

   (b) Remplaçons $[1, 3]$ par $[0, 3]$. Est-il toujours vrai que le temps de traitement est $\Theta(N)$? Expliquez.

2. (1 1/2 points)

   Dans la méthode *Quicksort*, à chaque étape on divise la liste à triér en deux sous-listes ; on met l'une des deux sous-listes sur une pile, et on s'occupe de l'autre.

   (a) Pour minimiser la profondeur de la pile, faut-il mettre la sous-liste la plus courte sur la pile, ou la sous-liste la plus longue ?

   (b) Faites un dessin qui montre le processus en forme de hiérarchie : chaque fois que nous sous-divisons une sous-liste, nous descendons un niveau dans la hiérarchie. À l'aide de ce dessin, donnez une borne supérieure pour la profondeur de la pile en supposant qu'on se sert de la politique dans votre réponse à la question 2a. La preuve de votre borne supérieure peut se faire en utilisant une preuve par induction sur le niveau dans la hiérarchie.

3. (1 point)

   Weiss, Exercice 2.7, page 50 : partie (a) seulement, et seulement les cas (2), (3), (4) et (5). Expliquez vos réponses.

4. (1 point)

   Weiss, Exercice 2.2, page 50 : parties (c) et (d) seulement. Dans chacun des deux cas, si l'énoncé est vrai, donnez la preuve. Si l'énoncé est faux, donnez un contre-exemple simple.

5. (1 point)

   Les bornes de style $O(\ldots)$ et $\Omega(\ldots)$ sont surtout utiles dans un contexte où on ne connaît pas la valeur de la fonction $T(N)$: le processus qui nous intéresse est normalement compliqué, et on n'est pas capable de dire exactement quelle est la fonction.

   Question triviale : supposons que $T(N)$ est connue explicitement, par exemple

   $$T(N) = \exp(N^3)/log(N), \quad 2 \leq N.$$

   Trouvez une fonction $g(N)$ telle que $T(N) = O(g(N))$, $T(N) = \Omega(g(N))$, et $T(N) = \Theta(g(N))$. Donnez la preuve.

6. (1 point)
   Weiss, Exercice 4.7, page 161. Il n'est pas nécessaire de donner la preuve de votre réponse à la deuxième partie, *i.e.*, quand a-t-on l'égalité ?
   La profondeur d_1 d'un arbre avec $M = 1$ noeud est égal à zéro.
7. (1 point)
   Weiss, Exercice 6.8, page 263 : parties (a) et (c) seulement.
8. (1 point)
   Soit $T_i(m_j, N)$ le coût de résoudre un problème de taille N en utilisant la méthode m_j. Dans la notation du livre (et du cours), soit le coût moyen

   $$T_{avg}(N) = \frac{1}{n}\sum_{i=1}^{n} T_i(m_j, N),$$

   le coût dans le plus mauvais cas

   $$T_{worst}(N) = \max_{i=1,\ldots,n} T_i(m_j, N),$$

   et le coût dans le meilleur cas

   $$T_{best}(N) = \min_{i=1,\ldots,n} T_i(m_j, N),$$

   où n est le nombre de problèmes de taille N.
   Évidemment $T_{avg}(N) \leq T_{worst}(N)$: en effet

   $$T_{avg}(N) = \frac{1}{n}\sum_{i=1}^{n} T_i(m_j, N) \leq \frac{1}{n}\sum_{i=1}^{n}(\max_{i=1,\ldots,n} T_i(m_j, N)) = \max_{i=1,\ldots,n} T_i(m_j, N).$$

   De façon semblable, $T_{avg}(N) \geq T_{best}(N)$.
   Est-il possible que $T_{worst} = O(N^2 \log N)$ et $T_{worst} = \Omega(1)$, en même temps que $T_{avg} = O(N^2)$, $T_{avg} = \Omega(\log N)$, $T_{best} = O(N \log N)$ et $T_{best} = \Omega(N)$? . Donnez une preuve formelle que c'est impossible, ou bien donnez des fonctions T_{worst}, T_{avg} et T_{best} qui satisfont aux conditions énoncées.

À réaliser en équipes de 1 ou 2. À remettre le 1 juin, 2016, avant `9:00`. Les solutions seront affichées le 1 juin. Les devoirs en retard ne seront pas acceptés.