Lista zadań. Nr 7. 19 maja 2009

Algorytmy i Struktury Danych

IIUWr. II rok informatyki.

1. Niech $A = \{a_1, a_2, \ldots, a_n\}$ będzie zbiorem kluczy, (takim, że $\forall_{i=1,\ldots,n-1} a_i < a_{i+1}$), które chcemy pamiętać w słowniku stałym. Znamy także ciąg $p_1, \ldots, p_n$ prawdopodobieństw zapytania o poszczególne klucze. Przyjmujemy, że $p_1 + p_2 + \ldots + p_n = 1$, a więc do słownika nie będą kierowane zapytania o klucze spoza słownika. Chcemy zaimplementować słownik jako drzewo BST. Ułóż algorytm znajdujący takie drzewo, które minimalizuje oczekiwany czas wykonywania operacji na słowniku.

   Punktacja:

   - za algorytm działający w czasie $O(n^3)$ - 1pkt;
   - za algorytm działający w czasie $O(n^2)$ - 2pkt;

2. (1pkt) Uzasadnij stwierdzenie, że funkcje hashujące potrzebne w konstrukcji słownika stałego (podanej na wykładzie) mogą być efektywnie znalezione.

3. (1pkt) Rodzinę H funkcji hashujących ze zbioru kluczy U w zbiór indeksów $\{0, .., m-1\}$ nazywamy *uniwersalną* (lub inaczej 2-*uniwersalną*), jeśli

$$\forall_{x \neq y \in U} Pr_{h \in H}[h(x) = h(y)] \leq \frac{1}{m}.$$

   Rozważmy następującą rodzinę funkcji haszujących. Niech uniwersum $U = \{0, 1, 2, \ldots, n-1\}$ i niech $V = \{0, 1, 2, \ldots, m-1\}$ będzie zbiorem wartości funkcji, gdzie $m \leq n$. Niech ponadto $p > n$ będzie liczbą pierwszą. Funkcje $h_{a,b}$ definiujemy w następujący sposób:

$$h_{a,b}(x) = ((ax + b) \bmod p) \bmod m.$$

   Udowodnij, że rodzina

$$H = \{h_{a,b} \mid 1 \leq a \leq p-1,\ 0 \leq b \leq p-1\}$$

   jest 2-uniwersalna.

4. (2pkt) Na wykładzie został przedstawiony dowód tego, że po umieszczeniu n kluczy w n-elementowej tablicy haszującej (z nawlekaniem kluczy o tej samej wartości funkcji haszującej na listy) z dużym prawdopodobieństwem żadna lista nie będzie zawierać więcej niż $\log n / \log \log n$ kluczy, o ile do haszowania użyta zostanie losowa funkcja haszująca h.

   - Przypomnij dowód (albo przedstaw inny) tego faktu.
   - Tak mocnego faktu nie da się udowodnić, gdy h jest losowo wybraną funkcją z 2-uniwersalnej rodziny funkcji haszujących. Udowodnij, że w tym przypadku prawdopodobieństwo, że któraś list zawiera więcej niż $\sqrt{2n}$ kluczy jest nie większe od $1/2$.

5. (2pkt) Przeprowadź analizę zamortyzowanego kosztu ciągu operacji *insert, deletemin, decrease-key, meld, findmin* wykonywanych na kopcach Fibonacciego, w których kaskadowe wykonanie operacji *cut* wykonywane jest dopiero wtedy, gdy wierzchołek traci trzeciego syna.