Tema Curs Structuri de Date 2

Horjea Cosmin-Marian
Anul 1 - Grupa 143
Universitatea din Bucuresti

June 6, 2020

Question 1

Demonstrati ca un arbore binar care nu este plin nu poate corespunde unui cod optim. Pentru definitia unui cod optim, va rog sa consultati cursul despre coduri Huffman.

Costul unui arbore de codificare Huffman pate fi reprezentat astfel:

$$\sum_{c \in A} (adancime(c) \cdot frecventa(c)),$$

unde A este alfabetul.

Presupunem ca putem avea un arbore T necomplet si care corespunde unei codificari Huffman optime.
Dar daca T este neplin $\Rightarrow \exists$ un nod $v \in T$ care are un singur fiu. Astfel, daca il sterg pe v din T, toti descendentii din al caror stramos este v vor avea adancimea scazuta cu 1. Astfel $cost(T - v) < cost(T)$ ceea ce intra in contradictie cu presupunerea ca T este optim $\square$

Question 2

Explicati cum se poate modifica metoda de sortare quicksort pentru ca aceasta sa ruleze in cazul cel mai defavorabil (i.e., worst-case) in timp $\mathcal{O}(nlogn)$, presupunand ca toate numerele ce trebuie sortate sunt distincte.

Stim deja ca algoritmul *quicksort* se descurca cel mai bine atunci cand pivotul ales este un element care se afla in mijlocul vectorului sortat.

Aceasta proprietate este gasita la mediana unui vector.

Astfel putem sa alegem pivotul folosind algoritmul de gasire a medianei intr-un vector nesortat, invatat la curs, care ruleaza in $\mathcal{O}(n)$, iar relatia de recurenta devine:

$$T(n) = 2T(n/2) + \Theta(n)$$

Deoarece am ales mediana, este garantat ca impartim vectorul in doua jumatati, iar la fiecare pas aflam mediana si partitionam elemntul in $\Theta(n)$. Din toate acestea, conform *Teoremei Master* avem o complexitate de $\mathcal{O}(nlogn)$ $\square$

Question 3

Fie T un arbore binar de cautare si x un nod din arbore care are doi copii. Demonstrati ca succesorul nodului x nu are fiu stang, iar predecesorul lui x nu are fiu drept.
Demonstram ca succesorul nu are fiu stang:

Fie x un nod cu doi fii iar S succesorul lui x , $S > x$ deci S se afla in subarborele drept al lui x.
Presupunem ca S are un fiu stang, pe care il notam cu z. Deci $z < S$, dar ambele se afla in subarborele drept al lui x, de unde rezulta ca $z > x$, deci succesorul lui x este z, ceea ce contrazice presupunerea ca S este succesorul lui x.

Demonstram ca predecesorul nu are fiu drept:

Fie x un nod cu doi fii iar P predecesorul lui x , $P < x$ deci P se afla in subarborele stang al lui x.
Presupunem ca P are un fiu drept, pe care il notam cu y. Deci $y > P$, dar ambele se afla in subarborele stang al lui x, de unde rezulta $y < x$, deci predecesorul lui x este y, ceea ce contrazice presupunerea ca P este predecesorul lui x. $\square$

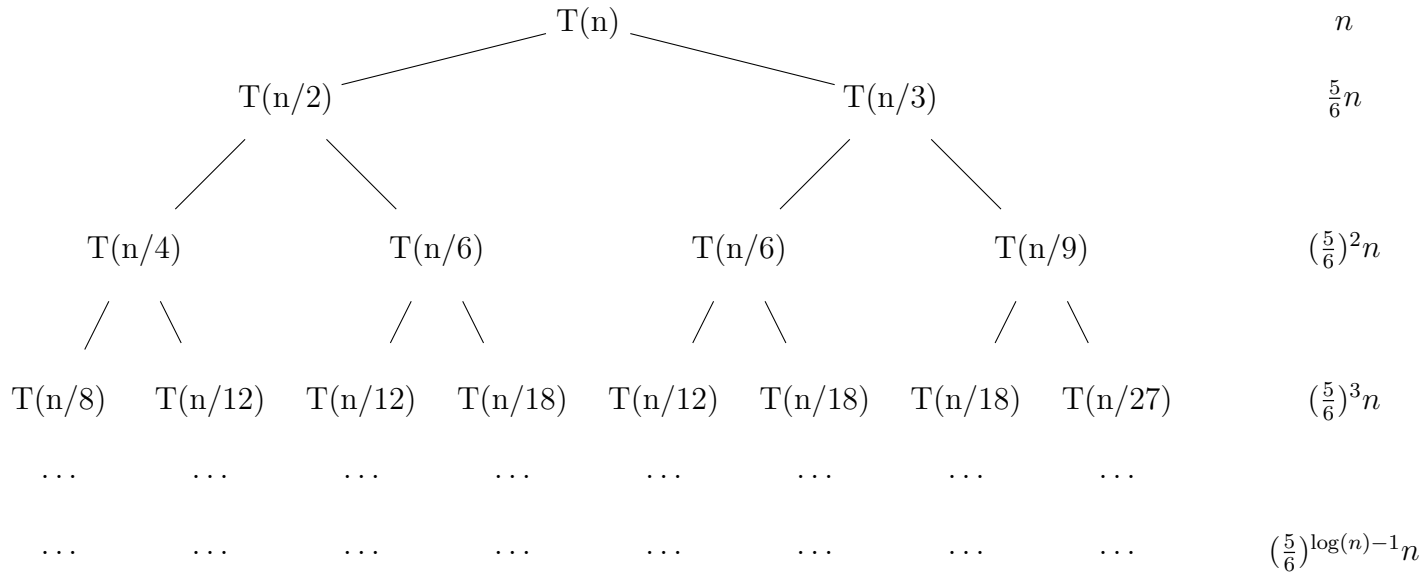
Question 4

Rezolvati recurenta $T(n) = T(n/2) + T(n/3) + n$. Demonstrati.

Folosim metoda iterativa:

$T(n) = T(n/2) + T(n/3) + n$
$T(n) = T(n/4) + T(n/6) + (n/2) + T(n/6) + T(n/9) + (n/3) + n$
$T(n) = n + (\frac{5n}{6}) + T(n/4) + 2T(n/6) + T(n/9)$
$T(n) = n + (\frac{5n}{6}) + T(n/4) + 2T(n/6) + T(n/9)$
$T(n) = n + (\frac{5}{6})n + T(n/8) + T(n/12) + \frac{n}{4} + 2(T(n/12) + T(n/18) + \frac{n}{6}) + T(n/18) + T(n/27) + \frac{n}{9}$
$T(n) = n + (\frac{5}{6})n + (\frac{5}{6})^2 n + T(n/8) + 3T(n/12) + 3T(n/18) + T(n/27)$
. . . . . .

Daca reprezentam arborele de derivare el va arata astfel:



Numarul de nivele: cel mult $\log(n)$
Numarul frunzelor din arbore: Pot sa fie maxim $2^{\log(n)} = n$, deci timpul de executie este $nT(1) = \mathcal{O}(n)$

Daca calculam costurile de pe fiecare linie a copacului, putem sa gasim o margine superiaora:

$$n + \left(\frac{5}{6}\right) n + \left(\frac{5}{6}\right)^2 n + \ldots \leq n \sum_{i=0}^{\infty} \left(\frac{5}{6}\right)^i = 6n = \Theta(n)$$

Total: $\Theta(n)$ $\square$