Dolne granice

niedziela, 16 czerwca 2024 10:59

Trudniej podać dolną niż górną granicę.

Dolna określa nam, że każdy algorytm rozwiązujący dany problem wykona NIE MNIEJ niż pewną liczbę operacji.

górna granica – $T(n)$ $\exists$ algorytm o złożoności $T(n)$

dolna granica – $T(n)$ $\forall$ algorytm o złożoności $T(n)$

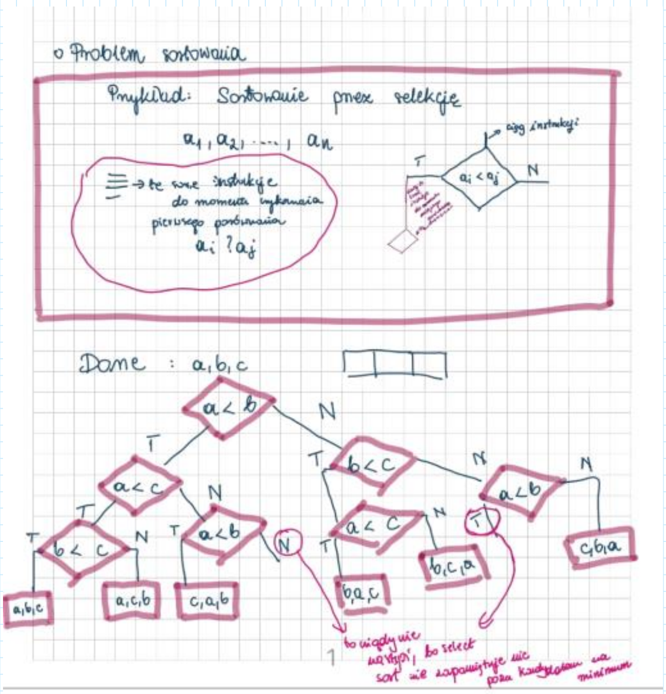
MODEL DRZEW DECYZYJNYCH

Comparison model – dane biorą udział tylko w porównaniach

Problem

Dane $a_1, \dots, a_n$

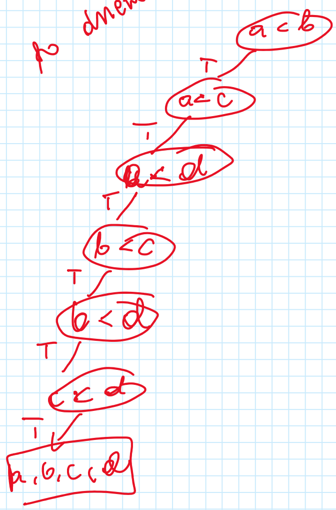
Zadanie:



Algorytm odpowiada modelowi drzew decyzyjnych

$\{D_i\}_{i=1}^{\infty}$

Załóżmy, że dla czteroelementowego zbioru to drzewo będzie wyglądać inaczej:



liczba liści w $D_n \geq n!$

↓

bo to liczba możliwych outputów sortowania a niektóre mogą się powtórzyć

Wiemy, że drzewo D_n jest drzewem binarnym

h_n – wysokość D_n

$2^{h_n} \geq n!$

$h_n \geq \lg n! \geq c \cdot n \log n$

Czyli każdy algorytm sortujący musi wykonać

A co jakbyśmy mieli drzewo trójarne? Nic to nie zmienia! Nadal mamy $n!$ liści, ograniczony jest przez

$3^{h_n} \geq n!$

$h_n \geq \log_3 n! = c \cdot \log_2 n!$

⇓

$n \log n$

$h_n \geq \log_3 n! = c \cdot \log_2 n!$

$\Downarrow$

$n \lg n$

Czyli każdy algorytm sortujący musi wykonać co najmniej $\Omega(n \log n)$ porównań

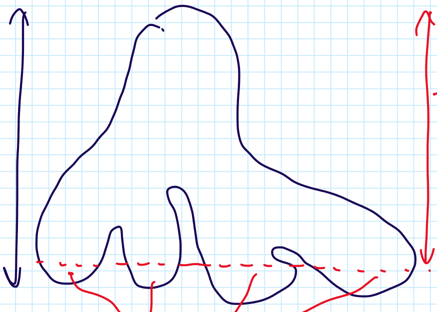
Ograniczenie na średnią głębokość:

Twierdzenie 2 *Jeżeli każda permutacja ciągu n-elementowego jest jednakowo prawdopodobna jako dana wejściowa, to wówczas każde drzewo decyzyjne sortujące ciągi n-elementowe ma średnią głębokość co najmniej $\log n!$.*

Uzasadnienie: Na głębokości nie większej niż $\log(n/e)^n - 1$ znajduje się mniej niż $n!/2$ liści. Tak więc co najmniej $n!/2$ liści osiągalnych z prawdopodobieństwem $1/n!$ leży na głębokości większej, co implikuje, że średnia wysokość drzewa decyzyjnego jest większa niż $(1/n!)(n!/2)\log((n/e)^n)$. □

Czyli średnie głębokości $\Omega(n \log n)$

$d = \lg n!$



$\rightarrow d-1 \Rightarrow$ będzie mniej $\frac{n!}{2}$ liści

$d = \lg \frac{n!}{2}$

bo na poziomie o jeden wyżej jest o połowę mniej liści czyli $\frac{n!}{2}$

$p_1 = p_2 = \ldots = p_{n!} = \frac{1}{n!}$

$\rightarrow \sum p_i d_i$

średnia wysokość drzewa

prawdopodobieństwo w ścieżce

długości ścieżki

Czyli dla $\frac{n!}{2}$ danych mamy ścieżki co najmniej d.

$\sum p_i d_i \geq \frac{n!}{2} \cdot d$

czyli średnie głębokości co najmniej $n \lg n$

Liniowe drzewa decyzyjne

Problem różnych elementów

Spojrzymy teraz nie na liczby jako na punkty $x_1, x_2, \ldots, x_n$ to teraz $(x_1, x_2, \ldots, x_n)$

punkt w R^n

FAKT: z każdym punktu spośród $P_1, \ldots, P_{n!}$ musimy dojść do innego liścia

załóżmy, że z P_i i P_j dochodzimy do tego samego liścia v

wyrazy

P_i x_1 x_r $\nearrow k$ .... x_k x_l ... x_n

P_j y_1 y_r ...... y_l ..... y_n

k k k

$t_j \; y_1 \; y_r \; \cdots y_v - \cdots y_n$

$y_r = k$, $y_v = k$

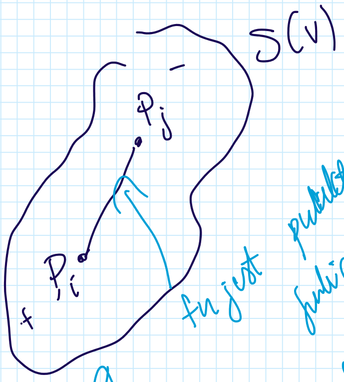
k - najmniejsze wartość, które jest na różnych miejscach w $x_1 \dots x_n$ i $y_1 \dots y_n$

$f(\langle x_1 \dots x_n \rangle) = x_r - x_v$

$f(P_i) > 0$

$f(P_j) < 0$

f jest ciągła



fn jest punkt gdzie funkcja przyjmuje wartość 0

Zbiór punktów, z których dochodzimy do V jest spójny