

ALGORYTMY I STRUKTURY DANYCH

WYBÓR k -TEGO ELEMENTU

Instytut Informatyki Uniwersytetu Wrocławskiego

Paweł Rzechonek

1. [***] Przedstawiony na wykładzie algorytm dokonuje jednoczesnego wyboru minimum i maksimum ze zbioru nieuporządkowanych elementów pamiętanych w tablicy. Określ ile porównań będzie wykonywał ten algorytm dla danych rozmiaru n . Jak bardzo różni się ten wynik od dolnej granicy $\lceil \frac{3}{2}n - 2 \rceil$?

```

function MINMAX-REC (ordtype  $T[0 \dots n - 1]$ )  $\mapsto$  (ordtype, ordtype)
{
    if  $n = 1$  then return  $(T_0, T_0)$ ;
    if  $n = 2$  then
        if  $T_0 \leq T_1$  then return  $(T_0, T_1)$ ; else return  $(T_1, T_0)$ ;
     $m \leftarrow \lfloor \frac{n}{2} \rfloor$ ;
     $(a, b) \leftarrow$  MINMAX-REC  $(T[0 \dots m - 1])$ ;
     $(c, d) \leftarrow$  MINMAX-REC  $(T[m \dots n - 1])$ ;
    return  $(\min\{a, c\}, \max\{b, d\})$ ;
}

```

Jak zapewne zauważyłeś/~aś, algorytm ten działa optymalnie dla danych, których rozmiar jest potęgą 2. Zmodyfikuj go w taki sposób, aby działał on optymalnie dla danych dowolnego rozmiaru.

2. [**] Wiemy, że dolna granica na liczbę wykonywanych porównań przez dowolny algorytm znajdujący minimum w n -elementowym zbiorze wynosi $n - 1$. Dolna granica na liczbę wykonywanych porównań przez dowolny algorytm znajdujący vice-minimum w n -elementowym zbiorze wynosi $n + \lceil \log n \rceil - 2$. Skonstruuj algorytm znajdujący vice-minimum w nieuporządkowanym n -elementowym zbiorze pamiętany w tablicy, działający optymalnie.
3. [*] Mamy dostęp do algorytmu *czarna skrzynka*, który w liniowym czasie znajduje medianę nieuporządkowanego ciągu liczb. W jaki sposób, korzystając z *czarnej skrzynki*, wyznaczyć dowolny k -ty co do wielkości element w nieuporządkowanym ciągu liczb. Twój algorytm powinien działać w czasie liniowym.
4. [**] Udowodnij, że w modelu drzew decyzyjnych wyznaczenie k -tego co do wielkości elementu w nieuporządkowanym ciągu n -elementowym oraz wydzielenie z tego ciągu $k - 1$ elementów nie większych i $n - k$ elementów nie mniejszych od piwota wymaga wykonania $\Omega(\log k + \log \binom{n}{k})$ porównań.
5. [**] Opisz algorytm, który wyznaczy medianę spośród 5 elementów, wykonując w najgorszym przypadku co najwyżej 6 porównań.
6. [***] Algorytmu Hoare'a wyznacza k -ty co do wielkości element z n -elementowego zbioru, przy czym $1 \leq k \leq n$. Udowodnij, że oczekiwany czas działania tego algorytmu wynosi $O(n)$. Za operację dominującą przyjmij porównania elementów.
7. [**] Czas działania algorytmu magicznych piątek można oszacować zliczając porównania, które on wykonuje. Liczbę tych porównań można określić następującym wzorem:

$$T(n) \leq \begin{cases} \frac{3}{8}n^2 & \text{dla } n < 15 \\ 6\lfloor \frac{n}{5} \rfloor + T(\lfloor \frac{n}{5} \rfloor) + n + T(n - 3\lceil \frac{n-4}{10} \rceil) & \text{dla } n \geq 15 \end{cases}$$

Oszacuj z dokładnością do stałego czynnika liczbę wykonywanych przez ten algorytm porównań.

Komentarz. Przypomnienie algorytmu magicznych piątek. Dla małych danych ($n < 15$) wystarczy posortować połowę tablicy algorytmem bąbelkowym. Dla większych danych ($n \geq 15$) wykonujemy cztery kroki: (i) wyznaczmy mediany ciągów 5-elementowych za pomocą 6 porównań, (ii) rekurencyjnie wyznaczamy medianę spośród tych median, (iii) dokonujemy trójkpodziału względem wyznaczonej wartości (iv) i rekurencyjnie rozwiązujemy problem dla jednej z części.