Lista 5 zadanie 4

Wiktor Hamberger
308982

26 maja 2020

Twierdzenie 1. *Aby scalić dwa n–elementowe ciągi trzeba wykonać co najmniej* $2n - 1$ *porównań.*

Dowód. Rozważmy następującą grę między algorytmem a złośliwym adwersarzem:

- Sytuacja początkowa: adwersarz twierdzi, że zna takie ciągi $A = a_0 a_1 \dots a_{n-1}$ i $B = b_0 b_1 \dots b_{n-1}$, których scalenie będzie wymagało wykonania co najmniej $2n - 1$ porównań. Algorytm nie zna A i B, wie tylko, że liczą one po n elementów każdy.
- Cel gry:
  - algorytmu: scalenie A i B używając mniej niż $2n - 1$ ruchów;
  - adwersarza: zmuszenie algorytmu do zadania co najmniej $2n - 1$ zapytań.
- Ruchy:
  - algorytmu: porównanie a_i i b_j, dla $0 \leq i, j < n$, algorytm nie pyta o porównania wyrazów w tym samym ciągu, ponieważ chce zminimalizować liczbę zapytań, a ciągi są posortowane;
  - adwersarza: odpowiedź na zapytanie.
- Koniec gry następuje, gdy algorytm scali A i B.

Tezę twierdzenia udowodnimy, jeżeli pokażemy, że adwersarz niezależnie od algorytmu posiada strategię wygrywającą. Strategia dla adwersarza:

- z zadania wiemy, że:
  - chcemy mieć $2n$ zestawów danych,
  - każde porównanie wykonane przez algorytm eliminowało co najwyżej jeden zestaw;

- rozważmy możliwe ciągi wynikowe W, utworzone w następujący sposób:
  - $W_0 = a_0b_0a_1b_1 \dots a_{n-1}b_{n-1} = w_0w_1w_2 \dots w_{2n-1}$,
  - W_i dla $i > 0$ powstaje w wyniku zamiany miejscami w_{i-1} i w_i.

  Ciąg W_0 ma $2n$ elementów, więc dostępnych różnych zamian jest $2n - 1$. To daje nam $2n$ różnych ciągów W_i. Dla zobrazowania:

$$
\begin{aligned}
W_0 &= a_0b_0a_1b_1 \dots a_{n-1}b_{n-1} \\
W_1 &= b_0a_0a_1b_1 \dots a_{n-1}b_{n-1} \\
W_2 &= a_0a_1b_0b_1 \dots a_{n-1}b_{n-1} \\
W_3 &= a_0b_0b_1a_1 \dots a_{n-1}b_{n-1} \\
&\vdots \\
W_{2n-1} &= a_0b_0a_1b_1 \dots b_{n-1}a_{n-1}
\end{aligned}
$$

- Rozważmy dowolne zapytanie algorytmu „jakie jest a_i w stosunku do b_j”. Mamy kilka możliwości:
  - $i = j$, wtedy adwersarz odpowiada, że a_i jest mniejsze od b_j. Eliminuje to jeden przypadek, w którym a_i jest większe od b_j, czyli $w_{2i} = b_j$ i $w_{2i+1} = a_i$, co odpowiada ciągowi W_{2i+1}.
  - $i = j + 1$, wtedy adwersarz odpowiada, że a_i jest większe od b_j. Eliminuje to jeden przypadek, w którym a_i jest mniejsze od b_{i-1}, czyli $w_{2i-1} = a_i$ i $w_{2i} = b_{i-1}$, co odpowiada ciągowi W_{2i}.
  - $i > j + 1$, wtedy adwersarz odpowiada, że a_i jest większe od b_j. Nie eliminuje to żadnego z przypadków, ponieważ $i - j \geq 2$, a kolejne ciągi W to zamiany a_x z b_x lub b_x z a_{x+1}, więc a_i i b_j są zbyt odległe od siebie.
  - $i < j$, wtedy adwersarz odpowiada, że a_i jest mniejsze od b_j. Podobnie jak wyżej nie eliminuje to żadnego przypadku, ponieważ nigdy nie zamieniamy ze sobą a_x i b_y dla $x < y$.

  Z tego wynika, że dla każdego zapytania, usuniemy maksymalnie jeden z możliwych ciągów W, więc potrzeba co najmniej $2n - 1$ zapytań, żeby jednoznacznie odkryć W, a co za tym idzie scalić A i B.

$\square$