AiSD ćwiczenia lista 1

Oskar Bujacz

6 kwietnia 2020

1. Zadanie 6

Dany jest niemalejący ciąg n liczb całkowitych dodatnich $a_1 \leqslant a_2 \leqslant ... \leqslant a_n$. Wolno nam modyfikować ten ciąg za pomocą następującej operacji: wybieramy dwa elementy a_i, a_j spełniające $2a_i \leqslant a_j$ i wykreślamy je z ciągu. Ułóż algorytm obliczający, ile co najwyżej elementów możemy w ten sposób usunąć.

Rozwiązanie:

Pomysłem na rozwiązanie zadania jest następujący algorytm zachłanny:

```
function(A[1:n]):
        i=1, j=int(n/2)+1, counter=0
        while j<= n :
                if 2A[i]<=A[j]:
                        counter+=1
                        A[i]=-1, A[j]=-1
                        i+=1, j+=1
                else:
                        j+=1
        return counter
```

Poprawność algorytmu:

Chcemy pokazać, że dowolne rozwiązanie optymalne możemy sprowadzić do rozwiązania otrzymanego przez powyższy algorytm. Załóżmy, że istnieje rozwiązanie optymalne. Weźmy dowolne z nich, usuńmy k par i rozważmy te k par (a_i, a_j) po usunięciu k par. Niech n to będzie liczba elementów ciągu. Podzielmy te liczby na 2 zbiory rozłączne L i R spełniające $a_i \in L$, $a_j \in R$ oraz $2a_i \leqslant a_j$ dla każdych i, j z usuniętych par. Pokażemy, że:

$$\forall_i a_i \leqslant min(R) \tag{1}$$

$$L = \{a_1, a_2, a_3, ..., a_k\} \tag{2}$$

$$R = \{a_{l1}, a_{l2}, a_{l3}, ..., a_{lk}\}, \qquad k+1 \leqslant li \leqslant n \tag{3}$$

Dowód (1)

Załóżmy, że istnieje element a_s, $s > k$, który został przydzielony do zbioru R. Niech będzie on sparowany z $a_t > a_s$. Rozważmy drugą parę (a_p, a_q), taką, że $a_p < a_t$. Z tego warunku wiemy, że możemy połączyć w pary wyrazy (a_s, a_p) oraz (a_t, a_q). Zatem uzyskujemy pożądaną własność (1).

Dowód (2)

Załóżmy, że $\exists a_i$, $i < k$, $a_i \notin L$. Weżmy najmniejszy taki, taki element a_j, $i < j < k$. Z monotoniczności ciągu $\{a_n\}$, nierówność $a_j \leqslant a_k$, gdzie a_k to element sparowany z a_j, będzie spełniona również dla a_i, zatem możemy dodać a_i do zbioru, a usunąć z niego a_j. Zachowujemy k elementów i warunki zadania. Jeśli wciąż istnieje element niespełniający (2), to ponownie wykonujemy ten sam algorytm.

Dowód (3)

Z (1) wiemy, że R, zawiera tylko elementy $\forall_{i,j} a_j \geqslant a_i$. Jeśli istnieje k elementów ze zbioru R, których indeksy $< \lfloor n/2 \rfloor + 1$, to możemy je wszystkie zamienić na elementy $a_s \geqslant a_{\lfloor n/2 \rfloor + 1}$, ponieważ jest ich co najwyżej $\lfloor n/2 \rfloor > k$.

Dowód optymalności

Teraz pokażemy dowód indukcyjny, który zapewni nas, że znalezione rozwiązanie jest optymalne.

Baza indukcji Do a_1 zostanie dobrane najmniejsze a_j, $2a_1 \leqslant a_j$, takie, że $j \geqslant [n/2] + 1$, zatem baza jest spełniona.

Krok indukcyjny

Weźmy dowolne $i < k$ i załóżmy, że algorytm znalazł i par liczb (a_p, a_q) spelniające warunki 1-3 oraz, że a_p zostało sparowane z najmniejszym mozliwym a_q. Przeprowadźmy kolejny krok algorytmu, dla a_{i+1} zostanie znaleziony najmniejszy a_r, $r \leqslant k$ spełniający nierówność z zadania. Na mocy zasady indukcji algorytm znajduje optymalne rozwiązanie.
Algorytm zachłanny znajduje maksymalną możliwą liczbę par spełniających $2a_i \leqslant a_j$.

Złożność czasowa i pamięciowa

Złożoność czasowa, to $O(n)$, ponieważ w najgorszym przypadku wykonujemy $n/2$ razy pętlę *while*.
Złożoność pamięciowa to również $O(n)$, bo musimy zadbać o przechowywanie tablicy n elementowej.