

# Binarne wyszukiwanie po wyniku

Ewa GOŁASZEWSKA\*, Marcin WIERZBIŃSKI\*\*

\*Uczennica, XIV LO im. Stanisława Staszica w Warszawie  
\*\*Doktorant, Wydział Matematyki, Informatyki i Mechaniki, Uniwersytet Warszawski

Wyszukiwanie binarne to algorytm służący do odnalezienia określonego elementu w posortowanej tablicy. Jego zaletą jest to, że umożliwia znalezienie elementu w znacznie krótszym czasie niż w przypadku tradycyjnego przeszukiwania tablicy, polegającego na „przechodzeniu” po niej element po elemencie i porównywaniu napotykanym elementom z szukanym.

Załóżmy, że dana jest posortowana tablica  $[a_0, a_1, \dots, a_{n-1}]$ ,  $a_0 \leq a_1 \leq \dots \leq a_{n-1}$  oraz pewna wartość  $x$ . Celem algorytmu wyszukiwania binarnego jest znalezienie pozycji elementu o wartości  $x$  albo stwierdzenie, że ten element nie pojawia się w tablicy (wówczas algorytm zwraca  $-1$ ).

Przykładem może być tablica liczb  $T = [3, 4, 4, 10, 11, 17, 18, 19, 21]$ , dla której chcemy sprawdzić, czy zawiera element  $x = 17$ . Algorytm wyszukiwania binarnego działa w następujący sposób. Rozpoczynamy od wybrania elementu w środku tablicy – wśród wartości  $[a_0, a_1, \dots, a_8]$  tym elementem jest  $a_4 = 11$  (pozycje w tablicy indeksujemy od 0). Następnie porównujemy ten element z wartością 17. Ponieważ 11 jest mniejsze od 17, to szukany element na pewno nie znajduje się w lewej połowie tablicy – jeśli jest w tablicy, to będzie w jej prawej części. W kolejnym kroku ograniczamy się więc tylko do prawej połowy  $[a_5, a_6, a_7, a_8]$  – jej środkowym elementem jest  $a_6 = 18$  (równie dobrze moglibyśmy wziąć  $a_7$ , ponieważ tablica, do której się ograniczyliśmy, ma parzystą liczbę elementów). Porównując go z 17, stwierdzamy, że musimy szukać w lewej części prawej połowy. Wówczas zostaje nam tylko element  $a_5 = 17$ , czyli nasz  $x$  jest w tablicy na pozycji 5.

Zauważmy, że podobne podejście często stosujemy intuicyjnie, na przykład przy wyszukiwaniu słowa w słowniku. Mając do znalezienia słowo *Delta*, nie zaczynamy przeglądać słownika od pierwszej strony wyraz po wyrazie, tylko otwieramy go we w miarę losowym miejscu, patrzymy, jakie wyrazy się tam znajdują, a z tego już wiemy, czy *Delta* jest na wcześniejszych, czy na dalszych stronach.

Algorytm wyszukiwania binarnego można zapisać w postaci pseudokodu następująco:

---

## Algorytm. Wyszukiwanie binarne

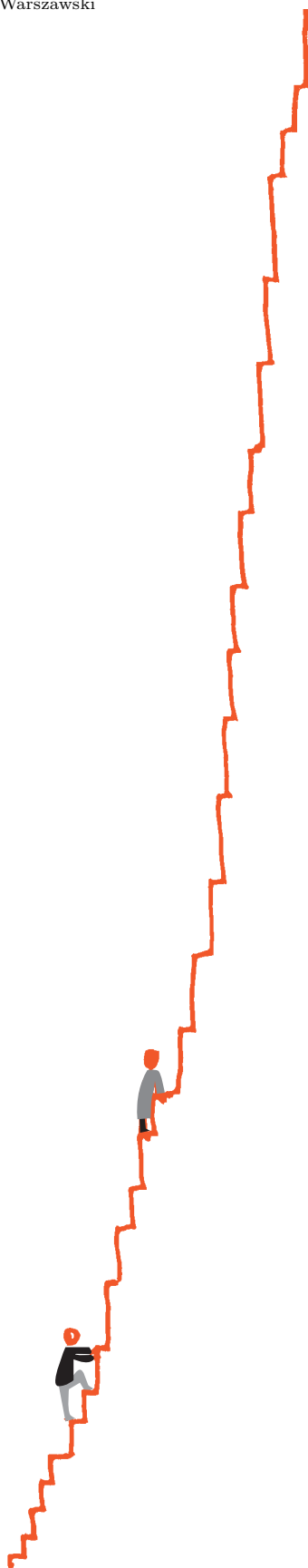
---

```
function BINARYSEARCH( $T, x, n$ )  
    lewy  $\leftarrow 0$   
    prawy  $\leftarrow n - 1$   
    while lewy  $\leq$  prawy do  
        środek  $\leftarrow \lfloor \frac{\text{lewy} + \text{prawy}}{2} \rfloor$   
        środkowaWartość  $\leftarrow T[\text{środek}]$   
        if środkowaWartość =  $x$  then  
            return środek ▷ znaleźliśmy  $x$   
        else if środkowaWartość <  $x$  then  
            lewy  $\leftarrow$  środek + 1 ▷ sprawdź w prawej połowie  
        else  
            prawy  $\leftarrow$  środek - 1 ▷ sprawdź w lewej połowie  
    return -1 ▷ nie udało się znaleźć.
```

---

Tak jak zauważyliśmy w przykładzie, porównując jakiś element  $a_i$  z szukaną wartością  $x$ , dowiadujemy się, czy szukany element jest na lewo, czy na prawo od  $a_i$ . Pozwala to w każdym kroku wyeliminować połowę elementów tablicy, a więc otrzymujemy złożoność czasową  $O(\log n)$ . Tradycyjne przeszukiwanie tablicy, przeglądające ją element po elemencie, w najgorszym przypadku może wymagać aż  $O(n)$  porównań.

Intuicyjnie jest jasne, że wyszukiwanie binarne działa poprawnie. Formalne udowodnienie poprawności algorytmu wyszukiwania binarnego wymaga jednak



odwołania się do pojęć takich jak „niezmiennik pętli”, ale w tym krótkim artykule nie będziemy o nich mówić.

Przykładem wykorzystania wyszukiwania binarnego jest zadanie „Wieża”, które pochodzi z Obozu Naukowego PROSERWY 2010, a jego autorem jest Jacek Tomasiewicz.

**Zadanie „Wieża”:** W Bajtocji wybudowano wysoką wieżę. Wejście na wieżę składa się z  $m$  schodków, a każdy schodek ma pewną wysokość. Bajtocką wieżę chce odwiedzić  $n$  mieszkańców. To, czy dana osoba jest w stanie pokonać kolejne schodki, zależy od jej wzrostu. Aby mieszkaniec Bajtocji mógł wejść na pewien schodek, musi być wyższy od schodka. Jeśli pewien schodek jest nie do przejścia przez mieszkańca, to zatrzymuje się on w danym miejscu na wieży – wyżej nie będzie mógł wejść. Znając wysokości kolejnych schodków i osób zwiedzających wieżę, chcielibyśmy wiedzieć, w którym miejscu zatrzyma się każdy mieszkaniec Bajtocji.

Zachęcamy Czytelnika do próby samodzielnego rozwiązania powyższego zadania przed przeczytaniem naszego rozwiązania.

**Rozwiązanie:** Aby rozwiązać zadanie, dla każdego schodka  $s$  najpierw obliczymy wysokość najwyższego schodka leżącego na drodze od spodu wieży do samego  $s$  (np. dla kolejnych wysokości schodków 1, 5, 3, 6, 2, 1, 7 stworzymy ciąg 1, 5, 5, 6, 6, 6, 7). W tym celu przechodzimy tablicę od pierwszego do ostatniego elementu, co krok aktualizując największą znaną dotychczas liczbę poprzez porównanie jej z wysokością aktualnego schodka. Otrzymujemy w ten sposób w czasie  $O(m)$  niemalejący ciąg. Zauważmy, że mieszkaniec o wzroście  $h$  wejdzie na schodek  $s$  tylko wtedy, gdy najwyższy schodek na drodze od spodu wieży do  $s$  jest niższy niż  $h$ . W związku z tym dla każdego mieszkańca możemy w czasie  $O(\log m)$  znaleźć miejsce, w którym się on zatrzyma – będzie to pierwszy schodek nie niższy od jego wzrostu. Całe rozwiązanie działa więc w czasie  $O(m + n \log m)$ .



## Zadania

Przygotował Dominik BUREK

**M 1771.** W liczbie naturalnej  $A$  przestawiono cyfry i otrzymano liczbę  $B$ . Wiadomo, że

$$A - B = \underbrace{11 \dots 1}_N$$

dla pewnej liczby całkowitej dodatniej  $N$ . Znaleźć możliwie najmniejszą wartość  $N$ , dla której takie liczby  $A, B$  istnieją.

Rozwiązanie na str. 5

**M 1772.** Na płaszczyźnie dany jest zbiór punktów, z których żadne trzy nie leżą na tej samej prostej. Narysowano pewną liczbę odcinków o końcach w tych punktach. Wiadomo, że każda prosta, która nie przechodzi przez dane punkty, przecina parzystą liczbę narysowanych odcinków. Udowodnić, że każdy punkt jest końcem parzystej liczby odcinków.

Rozwiązanie na str. 6

**M 1773.** Dany jest ostrosłup  $SA_1A_2 \dots A_n$ , którego podstawą jest wypukły wielokąt  $A_1A_2 \dots A_n$ . Dla każdego  $i = 1, 2, \dots, n$  w płaszczyźnie podstawy ostrosłupa konstruujemy trójkąt  $X_iA_iA_{i+1}$  przystający do trójkąta  $SA_iA_{i+1}$  i leżący po tej samej stronie prostej  $A_iA_{i+1}$  co podstawa (przyjmujemy  $A_{n+1} = A_1$ ). Udowodnić, że skonstruowane trójkąty pokrywają w całości podstawę.

Rozwiązanie na str. 8

Przygotował Andrzej MAJHOFER

**F 1089.** Jak zmieni się prędkość propagacji fal na powierzchni jeziora, jeśli wodę zastąpimy rtęcią, a jak, gdyby przyspieszenie ziemskie  $g$  wzrosło czterokrotnie? Gęstość rtęci:  $\rho_{Hg} \approx 13,55 \text{ g/cm}^3$ .

Rozwiązanie na str. 9

**F 1090.** Napięcie powierzchniowe roztworu wody z mydłem graniczącego z powietrzem wynosi:  $\sigma = 3 \cdot 10^{-2} \text{ N/m}$ . Roztwór posłużył do „produkcji” baniek mydlanych. Jaki warunek musi spełniać temperatura powietrza  $T_b$  w bańce o promieniu  $r = 2 \text{ cm}$ , żeby zaczęła unosić się w powietrzu o temperaturze  $T = 300 \text{ K}$ ? Ciśnienie atmosferyczne  $p_0 = 10^5 \text{ N/m}^2$ .

Rozwiązanie na str. 7

BABY  
BABY  
UDAJ SIĘ

