

# ALGORYTMY I STRUKTURY DANYCH

## ZŁOŻONOŚĆ OBLICZENIOWA ALGORYTMÓW

Instytut Informatyki Uniwersytetu Wrocławskiego

Paweł Rzechonek

1. Napisz funkcję obliczającą silnię dla długiej liczby  $N$ . Jaka jest złożoność czasowa tej funkcji? Pamiętaj, że mnożenie długich liczb zajmuje więcej niż jedną jednostkę czasu. Jaki wpływ na czas działania algorytmu może mieć fakt, że mnożymy te liczby od najmniejszej do największej, albo na odwrót od największej do najmniejszej? A może potrafisz wyznaczyć inną kolejność mnożenia poszczególnych liczb przy obliczaniu silni, która da jeszcze lepszy wynik?
2. Przypomnij sobie algorytmy sortowania *InsertSort* i *SelectSort*. Oszacuj złożoność czasową tych algorytmów na podstawie wykonywanych porównań.
  - (a) Jaka jest pesymistyczna i optymistyczna złożoność czasowa tych algorytmów? Dla jakich danych algorytmy te działają najszybciej, a dla jakich najwolniej?
  - (b) Jaka jest oczekiwana liczba porównań elementów w algorytmie *InsertSort*.
  - (c) Jaka jest oczekiwana liczba zamian elementów w algorytmie *SelectSort*.
3. Do  $n$ -elementowej tablicy  $T$  losowo wpisano liczby ze zbioru  $\{0, 1\}$  (wpisywane liczby były wybierane z rozkładem jednostajnym). Następnie dla tej tablicy uruchomiono opisany poniżej algorytm *find-zero*, który sprawdza czy znajduje się w niej chociaż jedna wartość 0.

```

(1)   function find-zero (integer  $T[n]$ )  $\mapsto$  boolean
(2)   {
(3)     for  $i = 0 \dots n - 1$  do
(4)       if  $T[i] = 0$  then
(5)         return true;
(6)     return false;
(7)   }
```

Wyznacz niezmiennik pętli z linii 3 i na jego podstawie uzasadnij poprawność algorytmu. Następnie oszacuj, jaki jest oczekiwany czas działania tej funkcji (za operację podstawową przyjmij porównanie z linii 4).

4. Do  $n$ -elementowej tablicy  $T$  zostały losowo wpisane liczby ze zbioru  $\langle 0, 1 \rangle$  (wpisywane liczby były wybierane z rozkładem jednostajnym). Następnie dla tej tablicy uruchomiono opisany poniżej algorytm *find-max*, który znajduje największą zapisaną w niej wartość.

---

```
(1)    function find-max (real  $T[n]$ )  $\mapsto$  real
(2)    {
(3)        real  $max \leftarrow T[0]$ ;
(4)        for  $i = 1 \dots n - 1$  do
(5)            if  $T[i] > max$  then
(6)                 $max \leftarrow T[i]$ ;
(7)        return  $max$ ;
(8)    }
```

---

Wyznacz niezmiennik pętli z linii 4 i na jego podstawie uzasadnij poprawność algorytmu. Następnie oszacuj, jaka jest oczekiwana liczba wykonywanych w linii 6 podstawień.

- (a) Jeśli liczba  $x$  jest losowo wybrana spośród  $n$  różnych liczb, to jakie jest prawdopodobieństwo, że  $x$  jest największą liczbą z tego zbioru?
- (b) Jakie jest prawdopodobieństwo, dla każdego  $i$  z zakresu  $0 \leq i < n$ , że jest wykonywana instrukcja podstawienia z wiersza 6?
- (c) Niech  $s_0, s_1, s_2, \dots, s_{n-1}$  będą  $n$  zmiennymi losowymi, gdzie  $s_i$  reprezentuje liczbę wykonań (0 lub 1) instrukcji podstawienia z wiersza 6 podczas  $i$ -tej iteracji pętli **for**. Ile wynosi  $E[s_i]$ ?
- (d) Niech  $s = s_0 + s_1 + s_2 + \dots + s_{n-1}$  będzie całkowitą liczbą wywołań instrukcji podstawienia z wiersza 6 podczas wywołania funkcji *find-max*. Pokaż, że  $E[s] \in \Theta(\log n)$ .