

ALGORYTMY I STRUKTURY DANYCH

METODA „DZIEL I ZWYCIĘŻAJ”

Instytut Informatyki Uniwersytetu Wrocławskiego

Paweł Rzechonek

1. [1 pkt.] Rozważmy rekurencyjny algorytm *sortowania głupiego* (stooge-sort):

```
procedure STOOGESORT (key  $T[0 \dots n-1]$ )
{
  if  $n < 6$  then { BUBBLESORT( $T$ ); return; }
   $p_1 \leftarrow \lfloor \frac{n}{3} \rfloor$ ;
   $p_2 \leftarrow n - p_1$ ;
  STOOGESORT( $T[0 \dots p_2-1]$ );
  STOOGESORT( $T[p_1 \dots n-1]$ );
  STOOGESORT( $T[0 \dots p_2-1]$ );
}
```

Uzasadnij poprawność tego algorytmu i oszacuj jego złożoność obliczeniową (skorzystaj z *uniwersalnego twierdzenia o rekurencji*).

2. [2 pkt.] Dane są dwie posortowane n -elementowe tablice liczb oraz liczba całkowita k z zakresu $0 \leq k < 2n$. Opisz algorytm znajdowania k -tej co do wielkości liczby spośród liczb zapisanych w obu tablicach. Czas działania twojego algorytmu powinien być rzędu $O(\log n)$. Uzasadnij jego poprawność.
3. [2 pkt.] Niech dane będą dwie długie liczby w systemie d -arnym: jedna o długości m i druga o długości n . Załóżmy, że $m \leq n$. Gdy $m \simeq n$, to algorytm pisemny oblicza iloczyn takich liczb w czasie $O(n^2)$, a algorytm z wykładu oparty na technice „dziel i zwyciężaj” potrzebuje tylko $O(n^{\log 3})$ czasu. Ale gdy m jest znacznie mniejsze niż n , to również ten ostatni algorytm jest nie do zaakceptowania. Skonstruuj algorytm, który dla przypadku gdy $m \ll n$ będzie działał w czasie $O(n m^{\log \frac{3}{2}})$.
4. [3 pkt.] Zmodyfikuj algorytm mnożenia długich liczb metodą „dziel i zwyciężaj” w ten sposób, aby liczba była dzielona na trzy części zamiast na dwie. Ile mnożeń krótszych liczb musisz wtedy wykonać? Jaka jest złożoność czasowa twojego algorytmu?
- Uwaga.* Asymptotyczny czas działania algorytmu z podziałem na trzy części powinien być lepszy niż z podziałem na dwie części.
5. [2 pkt.] Niech A będzie macierzą o wymiarach $k n \times n$ a B macierzą o wymiarach $n \times k n$, gdzie $k \in \mathbf{N}$. Jak szybko można obliczyć iloczyn $A \cdot B$ korzystając z algorytmu Strassena? O ile lepszy jest ten wynik od tradycyjnego mnożenia macierzy? Odpowiedz na te same pytania dla iloczynu $B \cdot A$.
6. [2 pkt.] Wyznacz wartość progową (*threshold*) opłacalności stosowania algorytmu Strassena w stosunku do tradycyjnego algorytmu mnożenia macierzy. Za operacje dominujące przyjmij działania arytmetyczne na elementach macierzy (dodawanie, odejmowanie i mnożenie pojedynczych liczb).