2010 Z Z9

Napisz procedurę partition (nie musi być to wersja z wykładu, ale musi być efektywna).

Pivot jest w A[p], funkcja zwaraca granicę podziału

```
Part(A[1..n], p, r)
  x <- A[p]
  i <- p-1

  j <- r+1

    while(i<j) do
      repeat(j--) until A[j] <= x
      repeat(i++) until A[i] >= x

      if(i<j) swap(A[i], A[i])
      else return j
```

2017 P Z7

Jaką złozoność ma algorytm Quicksort, w którym pivot wybierany jest algorytmem magicznych piątek?

Dzięki algorytmowi magicznych piątek uda nam się znaleźć mediane w czasie $T(n) = T(\frac{n}{5}) + T(\frac{7}{10}n) + O(n)$.

Stąd

$$T(n) = 2T(\frac{n}{2}) + T(\frac{n}{5}) + T(\frac{7}{10}n) + O(n)$$

$$T(n) >= 3T(\frac{n}{2}) + O(n) = O(n^{log_2 3})$$

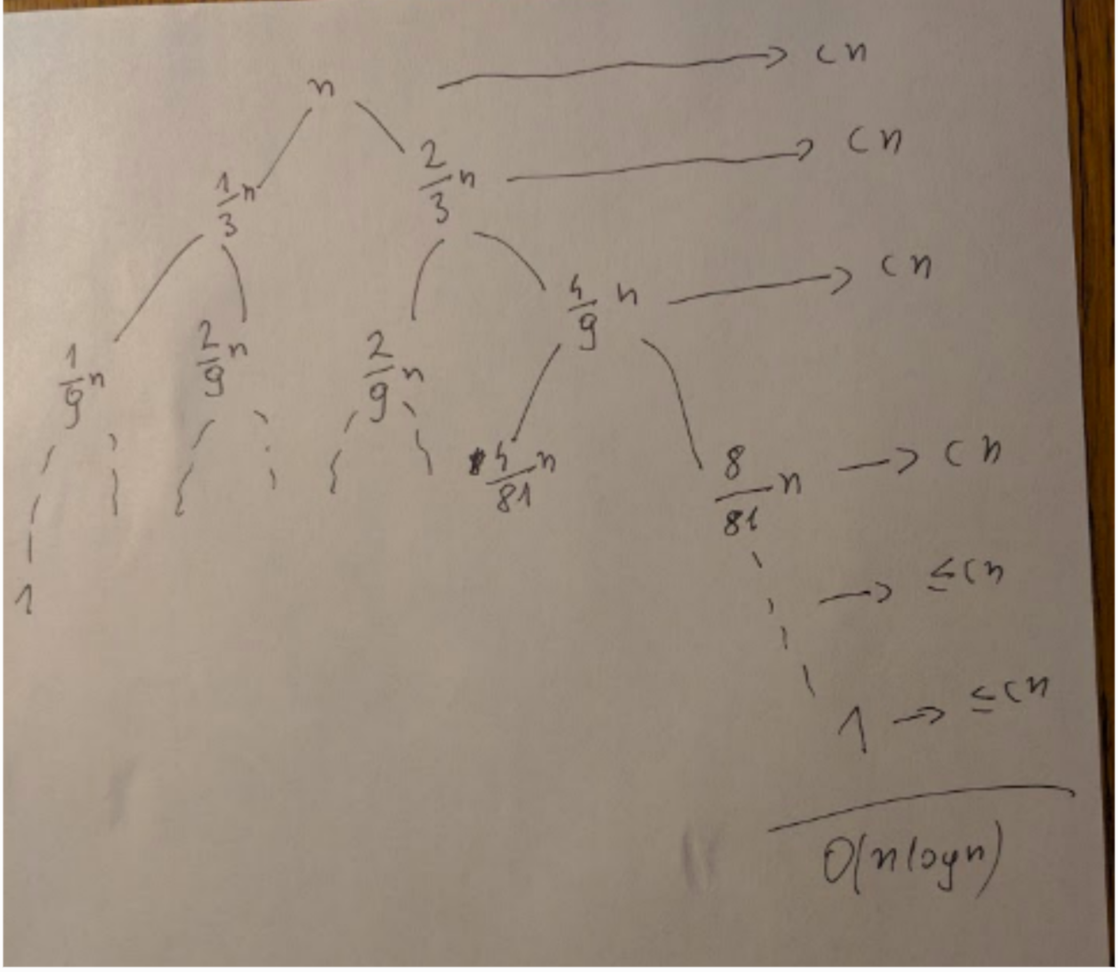
bardzo pesymistyczny scenariusz daje nam złożoność większą niż nlogn

2019 Z Z15

Zauważyłeś, że Quicksort (z deterministycznym pivotem) zachowuje się zadziwiająco regularnie na ciągach z pewnej rodziny A.

Otóż okazało się, że w trakcie wszystkich wywołań rekurencyjnych proderura Partition dokonuje podziału ciągu wejściowego na podciągi o długościach nie mniejszych niż 1/3 i nie większych niż 2/3 długości ciągu wejściowego.

W jakim czasie działa Quicksort na ciągach z rodziny A?



2019 P Z15

W analizie złożoności algorytmu QuickSort zakładaliśmy, że każde dwa elementy ciągu są porównywane ze sobą nie więcej niż jeden raz.

Zapisz w pseudokodzie procedury QuickSort i Partition realizujące tę własność.