

SELEKCIJA(IZBOR) | MEDIJAN

Mara Šumelj

Kolegij: Oblikovanje i analiza algoritama

Mentor: Prof.dr.sc. Saša Singer

Uvod

- Problem selekcije:
imamo skup A od n različitih brojeva te zadani broj k , takav da $1 \leq k \leq n$. Algoritam treba vratiti element skupa A koji ima rang k .
- Podsjetimo se, medijan niza je vrijednost za koju vrijedi da je 50% elemenata niza manje od ili jednako toj vrijednosti, a 50% veće od ili jednako njoj.
- Problem selekcije je generalizacija *problema medijana*.

Opis problema

- Ulaz : skup A od n različitih brojeva i $1 \leq k \leq n$
- Izlaz: k -ti element skupa A
- Najjednostavnije rješenje: sortirati A i vratiti k -ti element
- Složenost: $O(n \log n)$
- Možemo li bolje?

„The sieve technique”

- Tzv. „sito” tehnika
- Specijalni slučaj algoritma podijeli-pa-vladaj:
broj potproblema 1
- U svakom koraku eliminiramo dio elemenata za koje znamo da nisu tražena vrijednost

Selection algoritam

1. Korak: odabrati pivotni element
2. Korak: podijeliti skup A na 3 particije:
 - $A[q]$ - sadrži element x
 - $A[p, \dots, q-1]$ – sadrži sve elemente koji su manji od x
 - $A[q+1, \dots, r]$ – sadrži sve elemente koji su veći od x
3. Korak: ovisno o q završiti algoritam, odnosno iterirati na lijevo ili desno od q

Ilustracija algoritma selekcije

3
7
12
4
1
18
5

3
1
4
5
12
18
7

3
1
4

1
3
4

4

k=3
pivot=5

xRang=4
k=3

k=3
pivot=3

xRang=2
k=3-2=1

xRang=1
pivot=4

Pseudokod

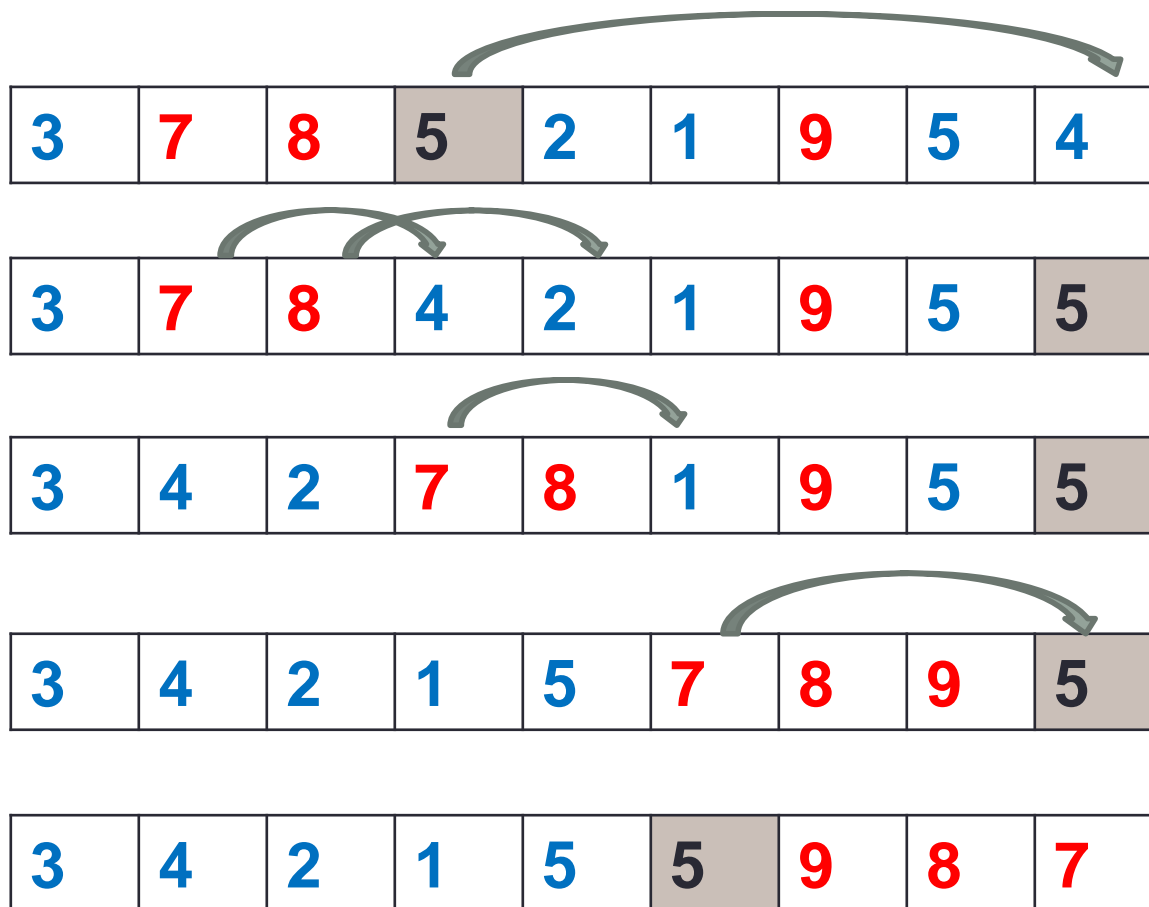
```
Selection(array A, int p, int r, int k){  
    if (p == r) return A[p]  
    x=PivotniElement(A, p, r)  
    q=Partition(A, p, r, x)  
    xRang = q - p - 1  
    if (k == xRang) return x  
    else if (k < xRang)  
        return Selection(A, p, q-1, k)  
    else  
        return Selection(A, q+1, r, k-xRang)
```

Algoritam selekcije

- 2 glavna problema su:
 - **1. Kako napraviti particiju niza?**
 - **2. Kako odabrati pivotni element?**

Particija niza

- Proces podjele na particije kao u Quicksort



Kako odabrati pivotni element?

- **Cilj:** eliminirati što veći dio niza
- Kandidati:
 - Nasumični broj
 - Srednji član niza
 - Medijan
- ✓ **Medijan medijana**

Medijan

- Medijan je „srednja točka” niza
- N neparan $\rightarrow m = \frac{n}{2}$
- N paran \rightarrow imamo 2 medijana
 - \rightarrow donji medijan $m = \left\lfloor \frac{n+1}{2} \right\rfloor$
 - \rightarrow gornji medijan $m = \left\lceil \frac{n+1}{2} \right\rceil$
 - \rightarrow ako nije drukčije specificirano, uzimamo donji medijan

Kako odrediti medijan medijana niza?

- Niz podijelimo u grupe po 5 elemenata
- Svakoj grupi odredimo medijan
- Pivot = medijan medijana

Medijan medijana

1	12	21	56	8	25	32
91	7	27	43	89	54	6
25	48	4	31	55	24	71
38	29	3	18	36	62	17
16	78	14	59	9	73	80



3	1	17	8	18	12	25
4	16	6	9	31	7	24
14	25	32	36	43	48	54
21	38	71	55	56	29	62
27	91	80	89	59	78	73

Medijan medijana

- najmanje je ranga $n/4$, a najviše $3n/4$.

3	1	17	8	18	12	25
4	16	6	9	31	7	24
14	25	32	36	43	48	54
21	38	71	55	56	29	62
27	91	80	89	59	78	73

Algoritam selekcije i medijan medijana

- Podijelimo niz A u grupe po 5 elemenata. Biti će točno $m = \left\lceil \frac{n}{5} \right\rceil$ grupa.
- U novi niz $median[]$ spremamo medijane tih grupa.
- Sortiramo svaku grupu i vratimo srednji element.
- Rekurzivno pozivamo algoritam selekcije na nizu medijana.

Analiza složenosti

- složenost algoritma ovisi o broju rekurzivnih poziva funkcije Selection().
- Složenost je u najgorem slučaju linearna!!!
- Funkcija Selection() će biti rekurzivno pozvana najviše $3n/4$ puta jer se funkcija Selection() rekurzivno poziva na nizu median koji ima $\lceil n/5 \rceil$ elemenata.

- $$T(n) \leq \begin{cases} 1 & \text{ako } n = 1 \\ T\left(\frac{n}{5}\right) + T\left(\frac{3n}{4}\right) + n & \text{inače} \end{cases}$$

Analiza složenosti

- Postoji konstanta c takva da $T(n) \leq cn$.
- Lako se dokaže indukcijom da za $c \geq \left(\frac{19c}{20}\right) + 1$ vrijedi tvrdnja.
- $T(n) \in O(n)$

Random_search

- Pivotni element = **random broj**
- Najgori scenarij: u svakom koraku izabere rubni element za pivota
- Složenost u najgorem slučaju: $O(n^2)$

Testiranje

- Algoritam je testiran na slučajno generiranim nizovima međusobno različitih cijelih brojeva
- Mjereno je prosječno vrijeme 10 pokretanja
- Jedinica je milisekunda

Testiranje

- Pivotni element = medijan medijana

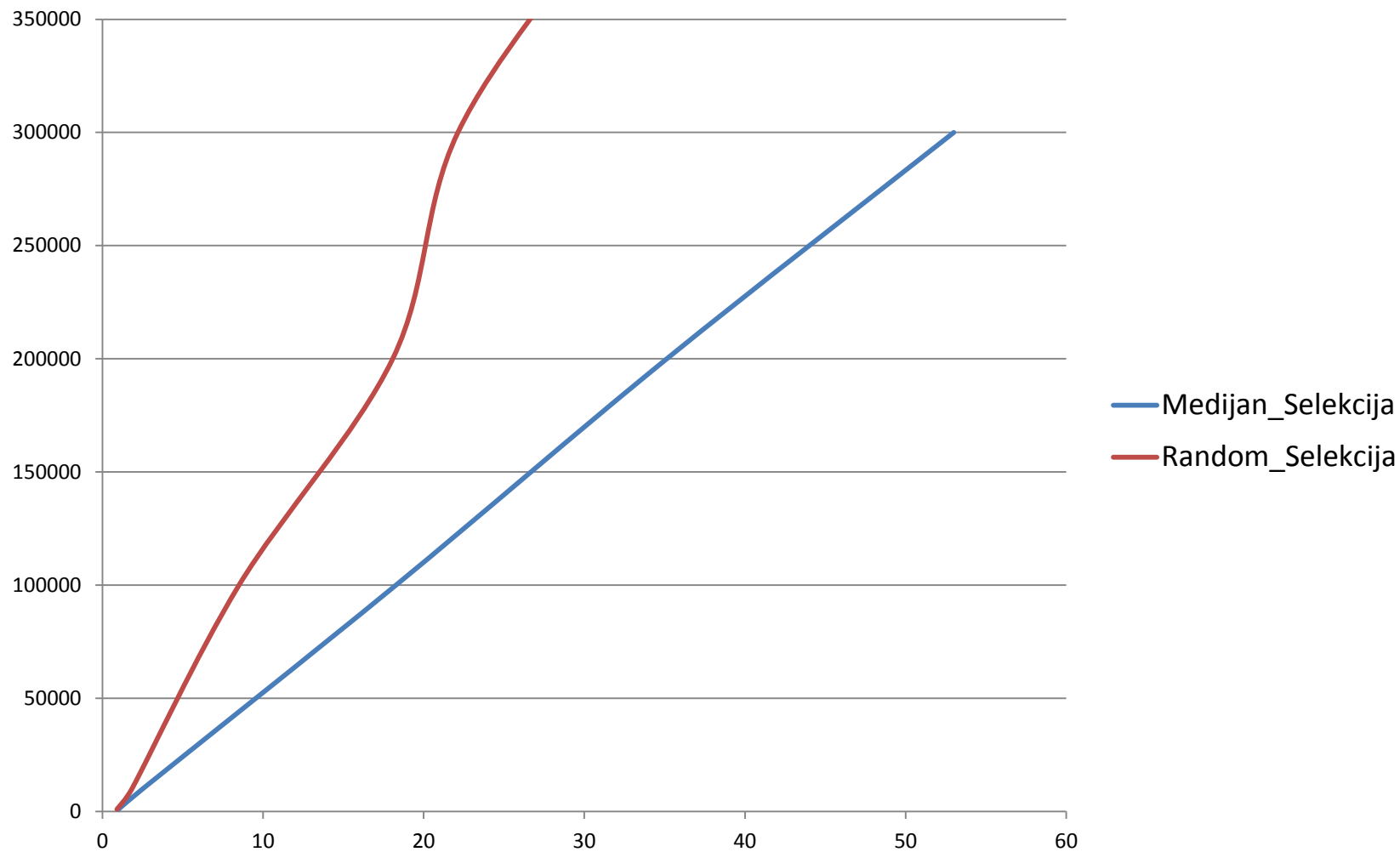
N	k=1	k=N/2	k=N	prosječno vrijeme
1 000	1,0	1,0	1,0	1,00
10 000	2,6	2,3	2,6	2,50
100 000	19,6	17,0	18,2	18,27
200 000	36,3	34,1	35,0	35,13
300 000	59,8	49,0	50,2	53,00

Testiranje

- Pivotni element = random element niza

N	k = 1	k = N/2	k = N	prosječno vrijeme
1 000	0,9	0,8	1,0	0,90
10 000	1,4	2,88	1,2	1,83
100 000	8,5	9,8	7,2	8,50
200 000	15,2	17,4	21,6	18,07
300 000	19,6	26,8	20,0	22,13
400 000	31,3	37,8	25,6	31,57
500 000	42,2	46,6	29,4	39,40

Random vs Median



Random vs Median

- Random_Selection $\rightarrow O(n^2)$
- Median_Selection $\rightarrow O(n)$
- A Random_Selection ipak u praksi daje bolje rezultate.
- Zašto?

Random vs Median

- $\frac{3n}{4} - \frac{n}{4} = \frac{n}{2}$

- Između $n/4$ i $3n/4$ gotovo pola elemenata niza
- Dakle u **50%** slučajeva, slučajnim odabirom eliminiramo $\frac{1}{4}$ niza

Literatura

- J. BENTLEY, *More programming pearls*, AT&T Bell Laboratories, Murray Hill, New Jersey, 1988., 159-170.
- T. H. CORMEN, C. E. LEISERSON, R. L. RIVEST, C. STEIN, *Introduction to Algorithms, 2nd edition*, MIT Press, Massachusetts, 2001., 160-172.
- D. M. MOUNT, *Design and analysis of computer algorithms*, Department of Computer Science, University of Maryland, 2008., 111-115.

Hvala na pažnji!