

# *Algoritmi sortiranja - Selection Sort - Zadaci*

## Istorija revizija

<i>Verzija</i>	<i>Autori</i>	<i>Datum</i>	<i>Status</i>	<i>Komentar</i>
1.00w	Marko Barjaktarović	18-Avg-2011	Inicijalna verzija	Početna verzija dokumenta
1.1	Saša Tošić	05-Mar-2014	Radna verzija	Izmenjeni su neki zadaci
1.2	Boris Šobot	10-Apr-2018	Radna verzija	Izmenjen jedan zadatak, promenjen redosled
1.3	Marko Vasiljević	22-Apr-2018	Radna verzija	Prilagođen online kursu Cosnovni

## Reference

## Sadržaj

Zadatak 1:	Prikaz sorta .....	5
Zadatak 2:	Broj pojavljivanja .....	5
Zadatak 3:	Tačke .....	5
Zadatak 4:	Optimizacija .....	5

## **Zadatak 1:     Prikaz sorta**

Napisati program koji sortira niz celih brojeva Selection Sort metodom i nakon svakog svakog traženja minimuma i eventualne zamene ispisuje ceo niz u posebnom redu. Takođe, ispisati i ukupan broj zamena koje je algoritam izvršio.

Primer. Za niz sa članovima 10, 4, 7, -1, 20 i 0, ispisuje se:

„Sortiranje niza izgleda ovako:

-1, 4, 7, 10, 20, 0

-1, 0, 7, 10, 20, 4

-1, 0, 4, 10, 20, 7

-1, 0, 4, 7, 20, 10

-1, 0, 4, 7, 10, 20

Ukupan broj zamena je 5.”

## **Zadatak 2:     Broj pojavljivanja**

Napisati program koji učitava niz od  $n$  elemenata, sortira ga od najmanjeg do najvećeg, a zatim štampa različite članove niza i pored svakog određen broj zvezdica u zavisnosti koliko se taj broj puta pojavljuje u nizu.

Primer. Za niz sa članovima 10, 7, 6, 8, 7, 8, 8, 11, 5 i 6, ispisuje se:

„5 \*

6 \*\*

7 \*\*

8 \*\*\*

10 \*

11 \*”

## **Zadatak 3:     Tačke**

Napisati program koji učitava  $n$  tačaka zadatih svojim (x, y) koordinatama u dva niza celih brojeva, a zatim sortira ove tačke po x-koordinati i ispisuje ih. Program napisati koristeći modifikaciju Selection Sort metode tako što će pri zameni mesta x koordinata dve tačke, biti zamenjene i y koordinate.

Primer. Ako su unete tačke (5, 5), (0, 0), (-3, 10), (10, 1) i (2, 6), tj. članovi niza x su 5, 0, -3, 10, 2, a članovi niza y 5, 0, 10, 1 i 6, ispisuje se: „(-3, 10) (0, 0) (2, 6) (5, 5) (10, 1)”

## **Zadatak 4:     Optimizacija**

Napisati program koji modifikuje Selection Sort algoritam tako da u svakom prolazu kroz segment niza traži ne samo najmanji nego i najveći element tog segmenta, pa najmanji stavlja na početak, a najveći na kraj.

Primer. Neka je unet niz sa članovima 10, 4, 7, -1, 20, 0. U prvom prolazu traži se najmanji element celog niza (-1) i najveći (20) i stavljaju se na početak, odnosno kraj; dobija se niz -1, 4, 7, 10, 0, 20. U drugom prolazu pretražuje se segment od drugog do pretposlednjeg elementa; najmanji element tog segmenta je 0 a najveći 10 pa se dobija niz -1, 0, 7, 4, 10, 20, itd. Na kraju programa, ispisuje se: „Sortiran niz izgleda ovako: -1, 0, 4, 7, 10, 20.”