

# *Algoritmi sortiranja - Insertion Sort - Zadaci*

## Istorija revizija

<i>Verzija</i>	<i>Autori</i>	<i>Datum</i>	<i>Status</i>	<i>Komentar</i>
1.00w	Aleksandar Manasijević	18-Apr-2018	Inicijalna verzija	Početna verzija dokumenta
1.1	Marko Vasiljević	23-Apr-2018	Radna verzija	Prilagođen online kursu Cosnovni
1.2	Saša Tošić	26-Apr-2018	Radna verzija	Dodati zadaci

## Reference

## Sadržaj

Prikaz sorta.....	5
Optimizacija.....	5
Granica.....	5
Najviše pojava.....	5
Sličice.....	6
Spajanje špilova.....	6
Prodavnica.....	6

## **Zadatak 1:     *Prikaz sorta***

**Opis:** Napisati program koji sortira niz celih brojeva Insertion sort metodom i nakon svakog prolaska kroz niz ispisuje ceo niz u posebnom redu. Takođe, ispisati i ukupan broj kopiranja podataka koje je algoritam izvršio.

**Primer.** Za niz sa članovima 10, 4, 7, -1, 20, 0 program ispisuje: „Sortiranje niza izgleda ovako:

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

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

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

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

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

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

Ukupan broj kopiranja podataka udesno je 19.“

## **Zadatak 2:     *Optimizacija***

**Opis:** Napisati program koji sortira niz celih brojeva optimizovanom Insertion sort metodom i nakon toga ispisuje sortirani niz redom. Optimizaciju izvršiti tako što će na početku najmanji broj biti stavljen na prvo mesto, a sortiranje počinje od trećeg elementa niza. Pri sortiranju, nije potrebno proveravati da li smo stigli do početka niza

**Primer:** Ako je dat niz 3, 8, 4, 2, 9 i 5, tada se prvo zamene brojevi 3 i 2 i dobijemo 2, 8, 4, 3, 9 i 5. Nakon toga, insert sort počinje da umeće brojeve od broja 4, jer su brojevi 2 i 8 sortirani.

## **Zadatak 3:     *Granica***

**Opis:** Posle okružnog takmičenja iz informatike, komisija je odlučila da povuče granicu za prolaz između dva takmičara između kojih je razlika u bodovima najveća. Napisati program koji učitava broj takmičara  $n$  ( $n > 1$ ) i broj bodova za svakog od njih i ispisuje koliko takmičara je prošlo dalje. Ukoliko se najveća razlika javlja na više mesta, uzeti ono koje dalje pušta više takmičara.

**Primer.** Neka je unet niz sa bodovima 90, 50, 55, 100, 25, 80, 22, 99. Program ispisuje „Dalje je prošlo 6 takmičara.“ Objašnjenje: sortirani niz izgleda ovako: 100, 99, 90, 80, 55, 50, 25, 22; najveće razlike su između takmičara sa 80 i takmičara sa 55, kao i između takmičara sa 50 i takmičara sa 25 bodova, pa se granica za prolaz povlači tako da više takmičara ide dalje, dakle na 50 bodova.

## **Zadatak 4:     *Najviše pojava***

**Opis:** Napisati program koji učitava niz od  $n$  brojeva i ispisuje kojih brojeva ima najviše i koliko ih ima. Da bi brže proverio koliko puta se koji broj ponavlja, program treba najpre da sortira niz metodom InsertionSort.

**Primer.** Ako su članovi niza 5, 4, 10, 4, 0, 12, 14 i 4, program ispisuje „Brojeva 4 ima 3 komada.“

## **Zadatak 5: Sličice**

**Opis:** Marko i Saša skupljaju Paninijeve sličice fudbalera. Svako je skupio po određeni broj sličica i hoće da se menjaju. Sašu interesuje koliko njegovih sličica Marko nema.

**Primer:** Ako Marko ima sličice 2, 4, 6, 8 i 10, a Saša sličice 3, 4, 5 i 6, tada Marko nema dve njegove sličice.

## **Zadatak 6: Spajanje špilova**

**Opis:** Marko i Saša skupljaju Paninijeve sličice fudbalera. Svako je skupio po određeni broj sličica, ali je Marko odlučio da sve svoje sličice pokloni Saši. Napisati program koji spaja špilove Sašinih i Markovih sličica, ako se zna da oni svoje špilove drže sortiranim rastuće.

**Primer:** Ako je Sašin špil karata 2, 6, 7 i 9, a Markov 3, 5, 8, 10 i 12, tada posle spajanja Saša ima špil 2, 3, 5, 6, 7, 8, 9, 10, i 12.

## **Zadatak 7: Prodavnica**

**Opis:** U prodavnici u redu na kasi stoji  $n$  kupaca. Neka svaki od njih zna koliko kupaca neposredno ispred njega ima veći račun od njega. Ispisati redosled kupaca po veličini računa, sortirano od najmanjeg do najvećeg.

**Primer:** neka su brojevi kupaca sa većim računima redom 0, 0, 2, 0 i 1. Tada najmanji račun ima kupac 3, pa zatim kupac 1, pa 2, pa 5, pa kupac 4.