



SVEUČILIŠTE U ZAGREBU
Fakultet prometnih znanosti
Zavod za inteligentne transportne sustave
Vukelićeva 4, Zagreb, HRVATSKA



Računalstvo

Primjeri izrade C# programa

Doc. dr. sc. Edouard Ivanjko, dipl.ing.

Sadržaj

- Uvod
- Brojanje podataka unutar intervala vrijednosti
- Najmanja i najveća vrijednost



- Općenita procedura izrade programa
 - Proučavanje problema
 - Razrada ideje
 - Poznate i nepoznate vrijednosti
 - Predviđanje problema
 - Popis varijabli
 - Izrada pseudokôda
 - Izrada i testiranje dijagrama toka
 - Raptor
 - Izrada i testiranje C# programa
 - MS Visual Studio

Uvod – Predviđanje problema

- Unos
 - Broj elemenata ima biti veći od nula
 - Smislenost vrijednosti fizikalnih veličina
 - Provjera ispravnosti specifičnih podataka
 - OIB, JMBAG, ...
- Matematičke operacije i funkcije
 - Dijeljenje s nulom
 - Korijen negativnog broja
- Obrada
 - Beskonačna petlja
 - Ispravan tip podatka za rezultat



Uvod – Popis varijabli

- Definirati
 - Količina potrebnih varijabli
 - Ime, tip i značenje varijable
 - Koristiti tip s najmanjom potrošnjom memorije
- Napisati u obliku tablice

Ime varijable	Tip varijable	Značenje varijable
brojiloVozilaD507	int	Brojilo vozila na državnoj cesti D507 između čvorova Valentinovo i Gubaševo na čvoru Krapinske Toplice
tempMotora	float	Temperatura motora broda izražena u stupnjevima Celsiusa
cijenaTaxiStart	float	Cijena početka taxi vožnje u kunama s uključenim PDV-om

Brojanje podataka unutar intervala vrijednosti

- Definicija problema

Potrebno je u skupu podataka prikazanih cijelim brojevima izbrojati sve podatke koji se nalaze unutar zadanog intervala. Donja granica intervala je uključena u skup podataka, a gornja granica nije. Broj pronađenih podataka je potrebno ispisati na zaslonu računala nakon što je operater unio sve podatke.



Brojanje podataka unutar intervala vrijednosti

- Proučavanje problema
 - Operater dobiva skup podataka
 - Operater traži broj podataka u nekom intervalu
 - Operater zna granice intervala podataka
 - Operater može izbrojati količinu podataka
 - Operater unosi podatke u računalo



Brojanje podataka unutar intervala vrijednosti

- Razrada ideje
 - Poznate i nepoznate vrijednosti
 - Količina podataka?
 - Zna operater
 - Sami podaci?
 - Zna operater
 - Broj podataka u nekom intervalu?
 - Traženi rezultat programa, sadržan unutar podataka
 - Predviđanje problema
 - Raspon granica intervala?
 - Granice mogu biti uključene ili isključene iz skupa podataka



Brojanje podataka unutar intervala vrijednosti

- Razrada ideje
 - Popis varijabli
 - Varijabla za unos količine podataka
 - Varijabla za unos pojedinog podatka
 - Varijabla za spremanje donje granice intervala
 - Varijabla za spremanje gornje granice intervala
 - Brojilo količine obrađenih podataka
 - Brojilo količine podataka u zadanom intervalu



Brojanje podataka unutar intervala vrijednosti

- Razrada ideje
 - Popis varijabli

Ime varijable	Tip varijable	Značenje varijable
brojPodataka	int	Količina podataka za obradu
podatak	int	Podatak koji je učitao za obradu
donjaGranica	int	Donja granica zadanog intervala podataka
gornjaGranica	int	Gornja granica zadanog intervala podataka
pronadeniPodaci	int	Broj podataka unutar zadanog intervala
obrađeniPodaci	int	Količina obrađenih podataka

Brojanje podataka unutar intervala vrijednosti

- Izrada pseudokôda

Deklaracija varijabli

brojPodataka, donjaGranica, gornjaGranica
obradeniPodaci, pronadeniPodaci, podatak

Inicijalizacija

obradeniPodaci := 0
pronadeniPodaci := 0

Unos

brojPodataka
donjaGranica
gornjaGranica



Brojanje podataka unutar intervala vrijednosti

- Izrada pseudokôda

Dok obradeniPodaci < brojPodataka Ponavlaj

Unos

podatak

Ako je (podatak \geq donjaGranica)

&& (podatak < gornjaGranica) **onda**

pronadeniPodaci := pronadeniPodaci + 1

obradeniPodaci := obradeniPodaci + 1

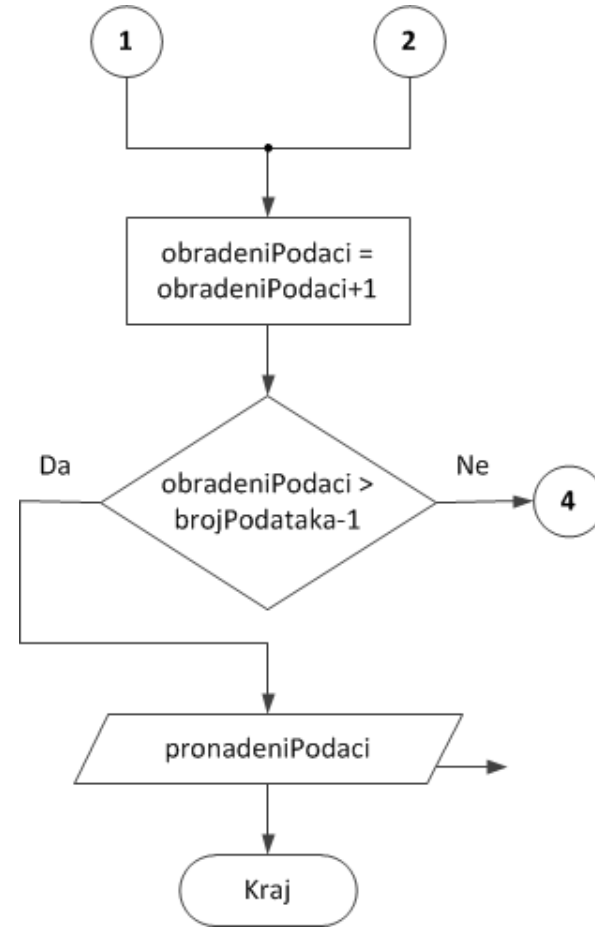
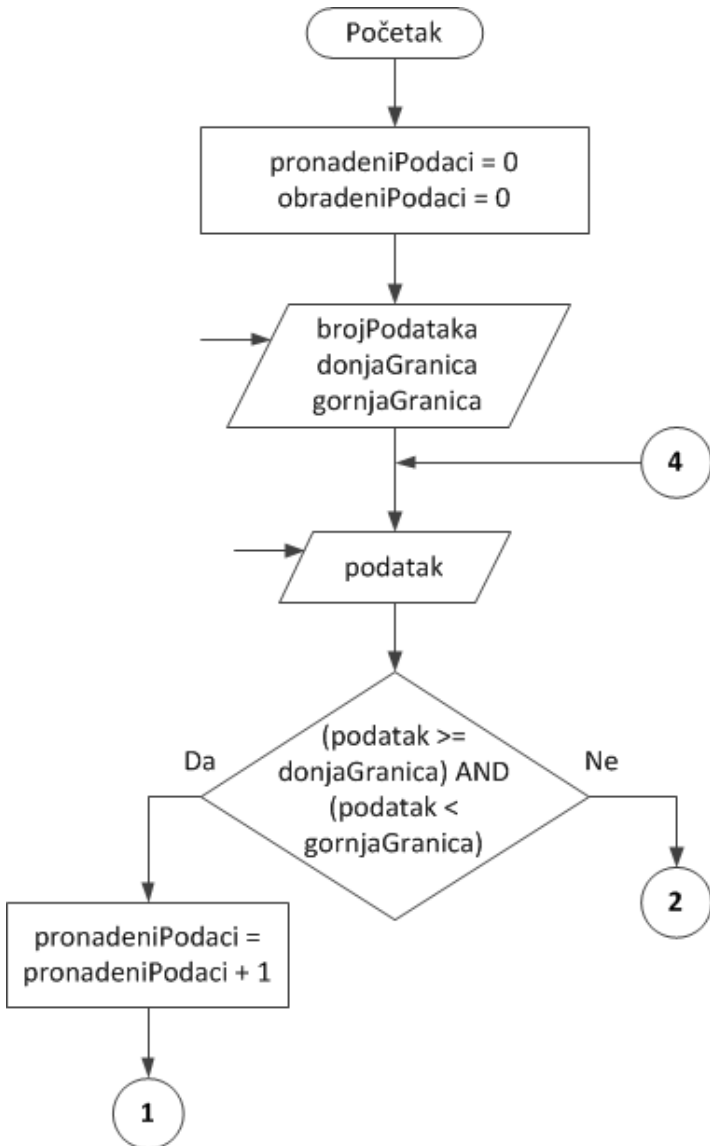
Ispis

“Broj podataka u zadanom intervalu je ” + pronadeniPodaci



Brojanje podataka unutar intervala vrijednosti

- Izrada dijagrama toka



 **IntervalVrijednosti**

Brojanje podataka unutar intervala vrijednosti

- Provjera dijagrama toka
 - Prolaz kroz dijagram toka i izvršavanje blokova
 - Označiti svaki blok svojom oznakom
 - Napraviti tablicu za izvršavanje
 - Redni broj koraka izvođenja
 - Oznaka bloka
 - » Pojedini blok može biti izvršen više puta
 - Rezultat izvršavanja bloka
 - Koristiti podatke iz definiranog skupa za provjeru
 - Svaki skup neovisno provjeriti
 - Koristiti samo dijagram toka bez pretpostavki
 - Usporediti rezultat izvođenja s točnim rezultatom
 - Razlika upućuje na pogrešku



Brojanje podataka unutar intervala vrijednosti

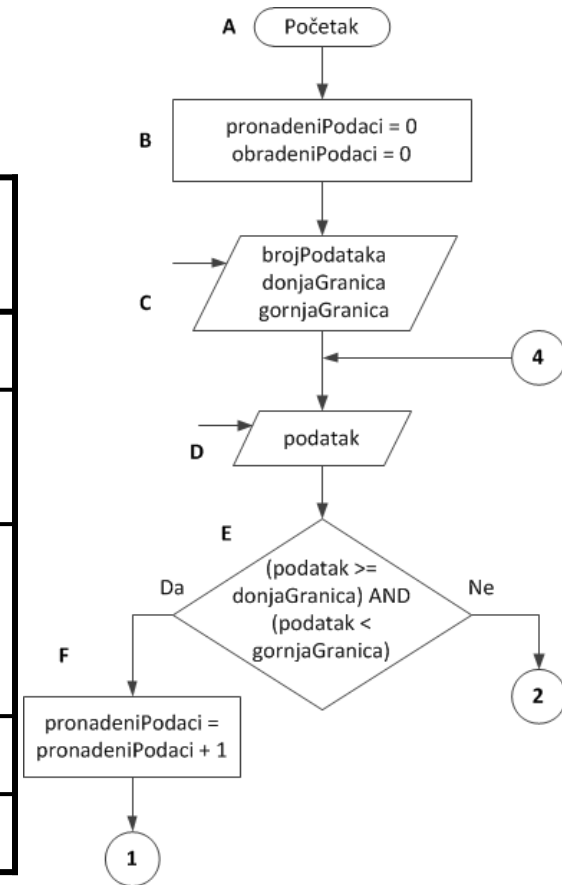
- Provjera dijagrama toka
 - Pokriti probleme primijećene u razradi ideje
 - Skup podataka za provjeru
 - Dovoljan jedan skup podataka
 - Ima sadržavati vrijednosti izvan intervala, unutar intervala i granice intervala
 - Primjer
 - Granice: 5 i 15
 - Vrijednosti: 4, 7, 15, 10, 16, -1, 0, 20, 14, 5, 9, 11
 - Rezultat: 6 vrijednosti unutar intervala



Brojanje podataka unutar intervala vrijednosti

- Provjera dijagrama toka
 - Koristi se definirani skup podataka

Korak izvođenja	Oznaka bloka	Rezultat izvođenja
1	A	Početak dijagrama toka
2	B	pronadeniPodaci poprima vrijednost 0 obradeniPodaci poprima vrijednost 0
3	C	brojPodataka poprima vrijednost 12 donjaGranica poprima vrijednost 5 gornjaGranica poprima vrijednost 15
4	D	podatak poprima vrijednost 4
5	E	uvjet poprima logičku vrijednost laž



– Analogno popuniti tablicu do kraja izvođenja

- Zbog petlje se neki blokovi izvedu više puta

Brojanje podataka unutar intervala vrijednosti

- Izrada C# programa

```
// deklaracija varijabli
int donjaGranica, gornjaGranica, brojPodataka, obradeniPodaci, pronadeniPodaci, podatak;

// unos i inijalizacija varijabli
Console.Write("Unesite broj podataka > ");
brojPodataka = Convert.ToInt32(Console.ReadLine());
Console.Write("Unesite donju granicu podatak > ");
donjaGranica = Convert.ToInt32(Console.ReadLine());
Console.Write("Unesite gornju granicu podatak > ");
gornjaGranica = Convert.ToInt32(Console.ReadLine());
pronadeniPodaci = 0;
obradeniPodaci = 0;

// učitavanje i obrada podataka
while(obradeniPodaci < brojPodataka)
{
    // unos podatka
    Console.Write("Unesite podatak > ");
    podatak = Convert.ToInt32(Console.ReadLine());

    // provjera intervala
    if ((podatak >= donjaGranica) && (podatak < gornjaGranica))
    {
        pronadeniPodaci = pronadeniPodaci + 1;
    }

    obradeniPodaci++;
}

// ispis rezultata
Console.WriteLine("Broj podataka unutar intervala je: " + pronadeniPodaci);
```

 **IntervalVrijednosti**

Brojanje podataka unutar intervala vrijednosti

- Provjera C# programa
 - Analogno provjeri dijagrama toka
 - Definirati skup podataka za provjeru
 - Iskoristiti isti skup podataka kao kod dijagrama toka
 - Označiti svaku liniju kôda oznakom
 - Napraviti tablicu za praćenje izvođenja kôda
 - Korak izvođenja, oznaka linije kôda, rezultat izvođenja
 - Pojedina linija kôda se može izvršiti više puta
 - Usporediti rezultate izvođenja s točnim rezultatom
 - Ponoviti za sve skupove podataka za provjeru
 - Ponekad je potrebno definirati više skupova



Najmanja i najveća vrijednost

- Definicija problema

Potrebno je u skupu podataka prikazanih cijelim brojevima pronaći najmanju i najveću vrijednost. Pronađene vrijednosti potrebno je ispisati na zaslonu računala nakon što je operater unio sve dostupne podatke.



Najmanja i najveća vrijednost

- Proučavanje problema
 - Operater dobiva skup podataka
 - Operater traži najmanju i najveću vrijednost unutar danog skupa podataka
 - Operater može izbrojati količinu podataka
 - Operater unosi podatke u računalo



Najmanja i najveća vrijednost

- Razrada ideje
 - Poznate i nepoznate vrijednosti
 - Količina podataka?
 - Zna operater
 - Sami podaci?
 - Zna operater
 - Najmanja i najveća vrijednost?
 - Traženi rezultat programa, sadržane unutar podataka
 - Predviđanje problema
 - Raspon vrijednosti u skupu podataka?
 - Nepoznat, može biti $(-\infty, \infty)$
 - Bitno kod inicijalizacije početne najmanje i najveće vrijednosti



Najmanja i najveća vrijednost

- Razrada ideje
 - Popis varijabli
 - Varijabla za unos količine podataka
 - Varijabla za unos pojedinog podatka
 - Varijabla za spremanje najmanje vrijednosti
 - Varijabla za spremanje najveće vrijednosti
 - Brojilo količine obrađenih podataka



Najmanja i najveća vrijednost

- Razrada ideje
 - Popis varijabli

Ime varijable	Tip varijable	Značenje varijable
brojPodataka	int	Količina podataka za obradu
podatak	int	Podatak koji je učitao za obradu
najmanjaVrijednost	int	Najmanja vrijednost unutar skupa podataka
najvecaVrijednost	int	Najveća vrijednost unutar skupa podataka
obrađeniPodaci	int	Količina obrađenih podataka

Najmanja i najveća vrijednost

- Izrada pseudokôda

Deklaracija varijabli

brojPodataka, podatak, najmanjaVrijednost
najvecaVrijednost, obradeniPodaci

Unos

brojPodataka
podatak

Inicijalizacija

najmanjaVrijednost := podatak
najvecaVrijednost := podatak
obradeniPodaci := 1



Najmanja i najveća vrijednost

- Izrada pseudokôda

Dok obradeniPodaci < brojPodataka **Ponavljaj**

Unos

podatak

Ako je podatak < najmanjaVrijednost **onda**

najmanjaVrijednost := podatak

Inače ako je podatak > najvecaVrijednost **onda**

najvecaVrijednost := podatak

obradeniPodaci = obradeniPodaci + 1

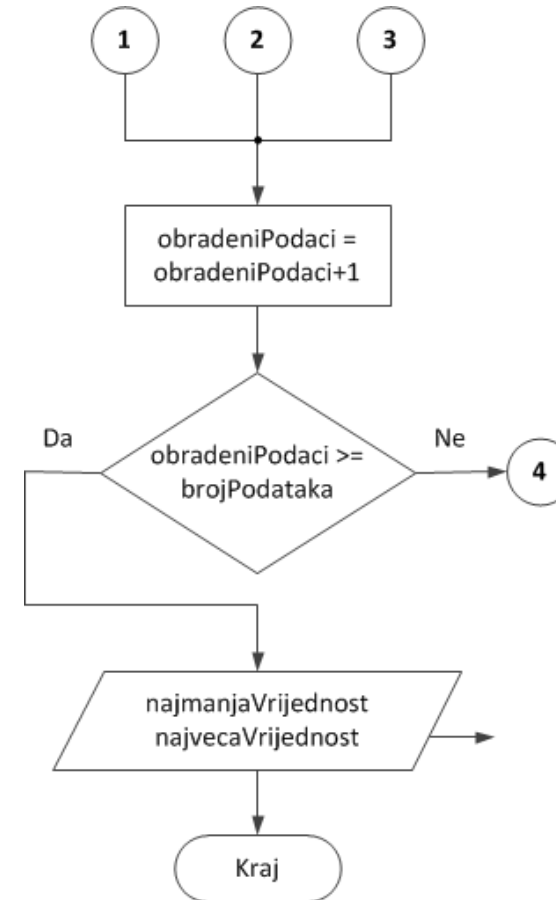
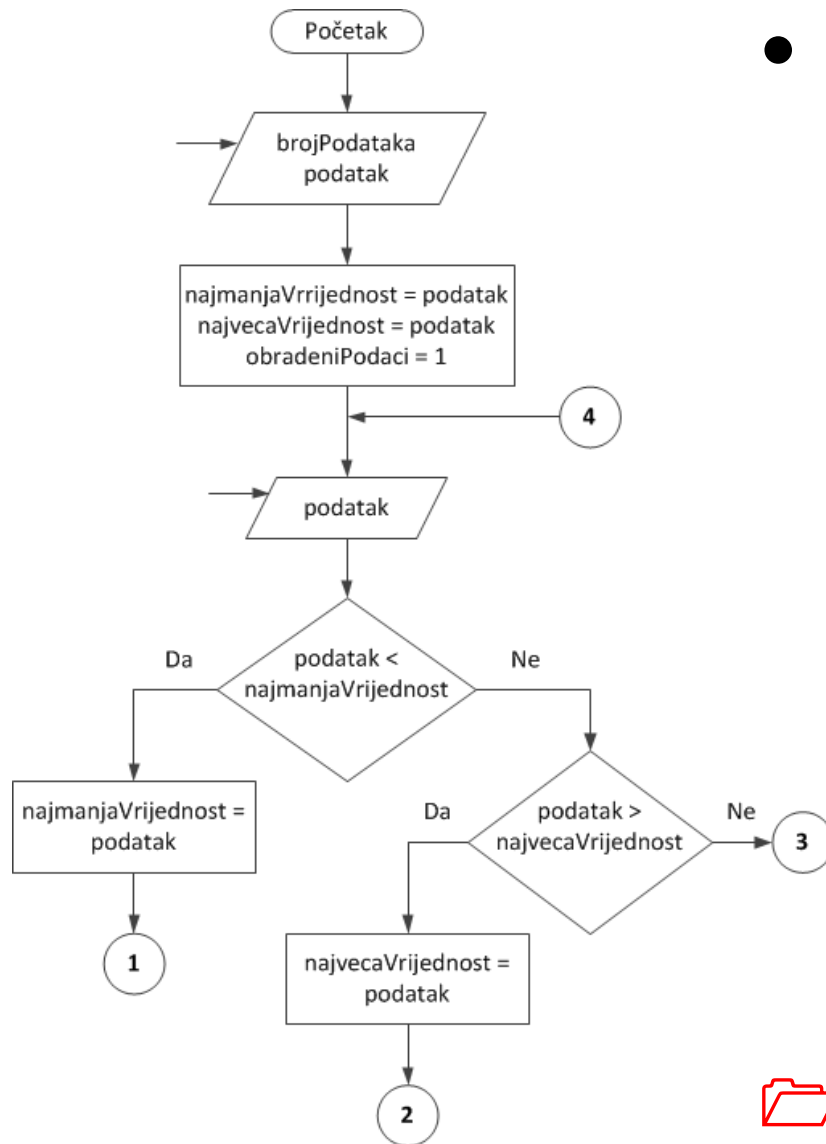
Ispis

"Najmanja vrijednost je " + najmanjaVrijednost

"Najveća vrijednost je " + najvecaVrijednost

Najmanja i najveća vrijednost

- Izrada dijagrama toka



 **NajmanjaNajvecaVrijednost**

Najmanja i najveća vrijednost

- Provjera dijagrama toka
 - Definiranje skupa podataka za provjeru
 - Pokriti probleme primijećene u razradi ideje
 - Potrebna tri skupa
 - Samo pozitivni brojevi bez nule
 - Samo negativni brojevi bez nule
 - Pozitivni i negativni brojevi
 - Primjer
 - Skup 1

5, 1, 8, 6	-> rezultat: 1 i 8
------------	--------------------
 - Skup 2

-5, -1, -8, -6	-> rezultat: -8 i -1
----------------	----------------------
 - Skup 3

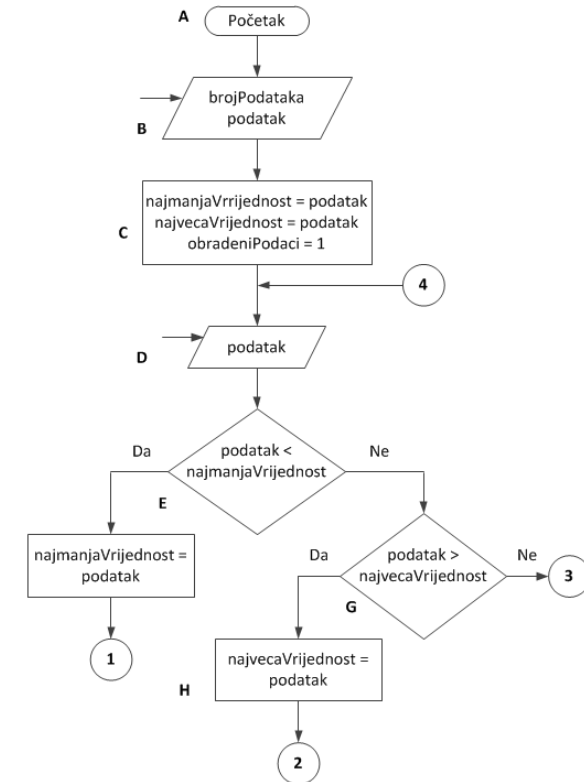
-5, 1, -8, 6	-> rezultat: -8 i 6
--------------	---------------------



Najmanja i najveća vrijednost

- Provjera dijagrama toka
 - Koristi se prvi skup podataka

Korak izvođenja	Oznaka bloka	Rezultat izvođenja
1	A	Početak dijagrama toka
2	B	brojPodataka poprima vrijednost 4 podatak poprima vrijednost 5
3	C	najmanjaVrijednost poprima vrijednost 5 najvecaVrijednost poprima vrijednost 5 obradeniPodaci poprima vrijednost 1
4	D	podatak poprima vrijednost 1
5	E	uvjet usporedbe istinit



- Analogno popuniti tablicu do kraja izvođenja
 - Ponoviti za drugi i treći skup podataka

Najmanja i najveća vrijednost

- Izrada C# programa

```
// deklaracija varijabli
int najvećiPodatak, podatak, brojPodataka, obradeniPodaci, najmanjiPodatak;

// unos i inijalizacija varijabli
Console.WriteLine("Unesite broj podataka > ");
brojPodataka = Convert.ToInt32(Console.ReadLine());
Console.WriteLine("Unesite prvi podatak > ");
podatak = Convert.ToInt32(Console.ReadLine());
najmanjiPodatak = podatak;
najvećiPodatak = podatak;
obradeniPodaci = 1;

// učitavanje i obrada ostalih podataka
while (obradeniPodaci < brojPodataka)
{
    Console.WriteLine("Unesite podatak > ");
    podatak = Convert.ToInt32(Console.ReadLine());
    if (podatak < najmanjiPodatak)
        najmanjiPodatak = podatak;
    else if (podatak > najvećiPodatak)
        najvećiPodatak = podatak;
    obradeniPodaci++;
}

// ispis rezultata
Console.WriteLine("Najmanja vrijednost je: " + najmanjiPodatak);
Console.WriteLine("Najveća vrijednost je: " + najvećiPodatak);
```

 **NajmanjaNajvecaVrijednost**

Najmanja i najveća vrijednost

- Provjera C# programa
 - Analogno provjeri dijagrama toka
 - Definirati skup podataka za provjeru
 - Iskoristiti iste skupove podataka kao kod dijagrama toka
 - Označiti svaku liniju kôda oznakom
 - Napraviti tablicu za praćenje izvođenja kôda
 - Korak izvođenja, oznaka linije kôda, rezultat izvođenja
 - Pojedina linija kôda se može izvršiti više puta
 - Usporediti rezultate izvođenja s točnim rezultatom
 - Ponoviti za sve skupove podataka za provjeru
 - Ponekad je potrebno definirati više skupova