

Dizajn i analiza algoritama

Lekcija 1: Uvod

leto 2019/2020

Prof. dr Branimir M. Trenkić

O meni....

- **Branimir M. Trenkić**
- Doktor tehničkih nauka, oblast – računarske i telekomunikacione mreže
- Redovni profesor **Fakulteta za kompjuterske nauke** “Megatrend” Univerziteta
- e-mail: trenkic.branimir@gmail.com
 - Molim da subject- linija Vašeg e-mail-a počinje sa kodom kursa (***DAA20***)

O kursu....

- Obim: **2 + 2**
- Termini:
 - **Predavanja: ponedeljak 13:00 – 14:45, K1**
 - Vežbe: termin – naknadno!
- Okosnica kursa:
 1. Prirodni ***nastavak SPA kursa***
 2. Dizajn i analiza algoritama (***kroz primere***)

O kursu....

- Način polaganja:
- ***Kroz predispitne obaveze:***
 - Odbrana vežbi (30 poena)
 - Aktivnost na nastavi (10 poena)
 - ***Teorija*** - Kolokvijum1 (30 poena) + Kolokvijum2 (30 poena)
 - U junskom roku je omogućeno kompletirati predispitne obaveze (***ali samo jedan deo!***)
- ***Završni ispit*** (u ispitnom roku)
 - Preduslov: Odbrana vežbi (30 poena)
 - Ispit se polaže pismeno (70 poena)
- Ispit je položen osvajanjem ***> 50 poena***

Osnovni materijal kursa

Dejan Živković

Uvod u algoritme i strukture podataka

Beograd, 2010

Poglavlja: 1, 2, 3, 5 i odeljak 4.2



Prezentacije predavanja (*pdf format*)

(Obavezna distribucija svim polaznicima kursa!)

Uvod

Niklaus Wirth:

ALGORITHMS + DATA STRUCTURES = PROGRAMS

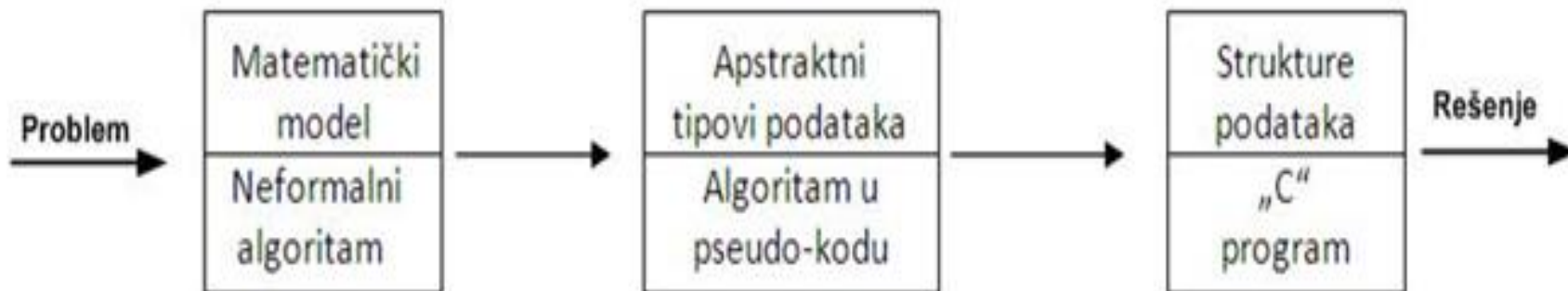
- **Algoritami i strukture podataka** – **fundamentalni predmet izučavanja** u računarstvu
- Računarski sistem:
 - **Smeštanje podataka** u memoriji računara
 - **Manipulacija podacima** iz memorije
 - Skup instrukcija na osnovu kojih računar vrši obradu podataka - **program**

Uvod

- U najopštijem smislu,
- **Struktura podataka** – termin koji opisuje **način organizacije podataka u programu**
- **Algoritam** – **postupak obrade podataka**
- **Strukture podataka + Algoritmi** – tesno povezani **gradivni elementi** od kojih se sastoji **program**
- **FKN** – 2 predmeta
 1. Strukture podataka i algoritmi
 2. **Dizajn i analiza algoritama**

Algoritmi

- Koriste se **za rešavanje računarskih problema** i u tesnoj su vezi sa strukturama podataka u tom kontekstu
- **Proces** rešavanja **računarskog problema**:



- Šta se podrazumeva pod računarskim problemom?

Algoritmi – zadavanje problema

- Da bi se proces rešavanja računarskog problema mogao uspešno okončati - **problem mora biti precizno definisan (zadat)**
- To je moguće uraditi na **dva načina**:
 1. Preciznom ***specifikacijom (opisom) ulazno-izlaznog odnosa*** (koji odslikava prirodu samog problema)
 2. U opštem slučaju koristi se ***formalniji (matematički) aparat*** – kako bi se otklonila svaka dvosmislenost

Algoritmi – zadavanje problema

- 1) Preciznom **specifikacijom (opisom) ulazno-izlaznog odnosa** kojim se izražava priroda problema
 - Problem **sortiranja niza brojeva**
 - **Ulaz** – niz brojeva u proizvoljnom redosledu
 - **Izlaz** – niz istih brojeva poređanih u rastući redosled
 - Problem **usitnjavanja novčanog iznosa**
 - **Ulaz** – novčani **iznos** i **apoeni** metalnih novčića
 - **Izlaz** – minimalni broj tih novčića kojima je moguće usitniti dati novčani iznos

Algoritmi – zadavanje problema

- Kod mnogih problema *nije dovoljno* na ovaj neformalni način *definirati ulazno-izlazne odnose*
 - Problem sortiranja
 - Da li neki od brojeva u ulaznom nizu *mogu biti jednaki* i kako ih poredati u izlaznom nizu?

Algoritmi – zadavanje problema

2) Formalni način definisanja (zadavanja) problema

Iz dva dela:

- 1. Ulazni parametri** i odgovarajući **izlazni parametri**
 - Obe vrste parametara zadovoljavaju **određene uslove** – kojima se definiše sam problem
- U literaturi se često koristi pojam **instanca problema** koji podrazumeva skup konkretnih vrednosti za ulazne parametre, a pod **rešenjem instance problema** – podrazumevaju se konkretne vrednosti izlaznih parametara

Algoritmi – zadavanje problema

B) Formalni način definisanja (zadavanja) problema

- Instanca problema sortiranja - ***[23,17,8,20,15]***
- Rešenje te instance - ***[8,15,17,20,23]***

Algoritmi – zadavanje problema

- Ako smo zadali računarski problem – sledi definisanje **matematičkog modela problema**
- Model se sastoji **iz dva dela** koji **opisuju** njegove **ulazne** i **izlazne** parametre
- Oba dela se predstavljaju **u vidu odgovarajućih matematičkih objekata** kao što su brojevi, skupovi, funkcije itd...
- **Ulazni deo** - **opšti model instance** problema
- **Izlazni deo** – **opšti model rešenja instance** problema

Algoritmi – zadavanje problema

Formalni način definisanja (zadavanja) problema

Ulaz: (*opšti model instance:*)

pozitivan **ceo broj** k (broj metalnih novčića)

niz od k celih brojeva $[c_1, \dots, c_k]$ (vrednost pojedinačnih apoenā)

a novčani iznos

Izlaz: (*opšti model rešenja instance:*)

niz od k celih brojeva $[b_1, \dots, b_k]$ takav da važi:

$$a = \sum_{i=1}^k b_i \cdot c_i \quad \sum_{i=1}^k b_i \quad \text{je minimalno}$$

Algoritmi – rešenje problema

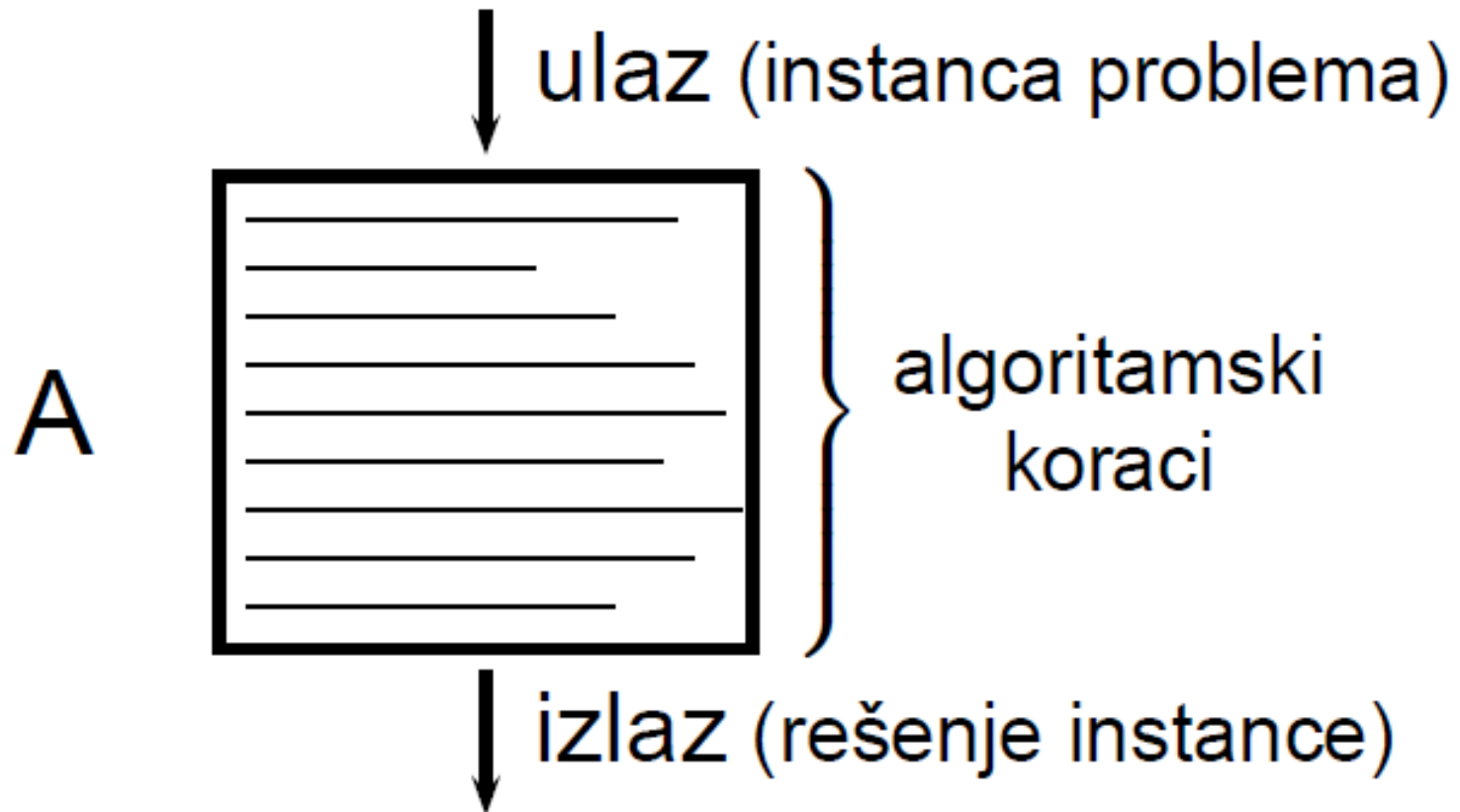
- **Nakon** kreiranja odgovarajućeg **matematičkog modela** za dati problem - treba **naći rešenje** na bazi definisanog modela
- Početni ili ***među-korak*** je **opisati rešenje** – **opis rešenja nazivamo algoritam**
- ***Šta je algoritam?***
- ***DEF.***- Algoritam je ***diskretan i jednoznačan postupak*** predstavljen ***konačnim*** brojem instrukcija (pravila) koji ***dovodi do rešenja*** nekog konkretnog ***problema***

Algoritmi

- Ili,
- **Algoritam** je tačno definisana **računarska procedura** koja pretpostavlja neke podatke kao **ulaz** i proizvodi neke podatke kao **izlaz**
- Sastoji se od **niza preciznih koraka** koji dovode do rešenja datog problema
- Svi koraci se mogu mehanički izvršiti na računaru

Algoritmi

- Analogija sa *kuhinjskim receptom*
- Algoritam \neq program



Zapis algoritma

- Postoji **više načina** kako se algoritam može **predstaviti**
- Željeni **stepen formalizma** u predstavljanju
 1. Korišćenje **govornog (prirodnog) jezika** u prikazu algoritma
 2. Korišćenjem nekog **programskog jezika** sa njegovim strogim sintaksnim i semantičkim pravilima
- Kompromisna rešenja:
 1. Algoritamske šeme kao grafički način i
 - 2. Pseudo jezici (ili, strukturni prirodni jezici)**

Zapis algoritma - *NZD*

„Еуклидов алгоритам је дека свих алгоритама, пошто је то најстарији нетривијални алгоритам који је преживео до данас.“

Доналд Кнут, *The Art of Computer Programming, Vol. 2: Seminumerical Algorithms*, 2nd edition (1981), стр. 318.

- ***NZD(x, y)***
- **Iterativni postupak** – rezultat iz **više koraka** (u svakom koraku obrada je **identična!**)
- Rezultat jednog koraka – koristi se u sledećem
 1. Postupak koji se ponavlja navodi se **u formi tela petlje**
 2. Broj ponavljanje se kontroliše **izlaznim uslovom petlje**

Zapis algoritma - ***NZD***

- ***NZD(x, y)*** – Euklidovo (iterativno) rešenje
- Rešenje se zasniva na ***jednakosti najvećeg zajedničkog delioca sukcesivnih ostataka***
- Svaki korak počinje sa dva ostatka (inicijalno a i b)
 $nzd(x, y) = nzd(y, x \% y)$
- Sve dok je drugi ostatak $\neq 0$
 $nzd(x, 0) = x$

Zapis algoritma - *NZD*

$$nzd(12378, 3054) = ?$$

- Rešenje:

$$nzd(x, y) = nzd(y, x \% y)$$

12378, 3054

12378, 3054, 162 (**12378 % 3054 = 162**)

12378, 3054, 162, 138 (**3054 % 162 = 138**)

12378, 3054, 162, 138, 24 (**162 % 138 = 24**)

12378, 3054, 162, 138, 24, 18 (**138 % 24 = 18**)

12378, 3054, 162, 138, 24, 18, 6 (**24 % 18 = 6**)

12378, 3054, 162, 138, 24, 18, 6, 0 (**18 % 6 = 0**)

⇒ **$nzd(12378, 3054) = 6$**

Zapis algoritma - *NZD*

- *NZD(a, b)* – *iterativno rešenje*

```
// Ulaz: pozitivni celi brojevi x i y
```

```
// Izlaz: nzd(x,y)
```

```
algorithm euklid(x, y)
```

```
    while (y > 0) do
```

```
        z = x % y;
```

```
        x = y;
```

```
        y = z;
```

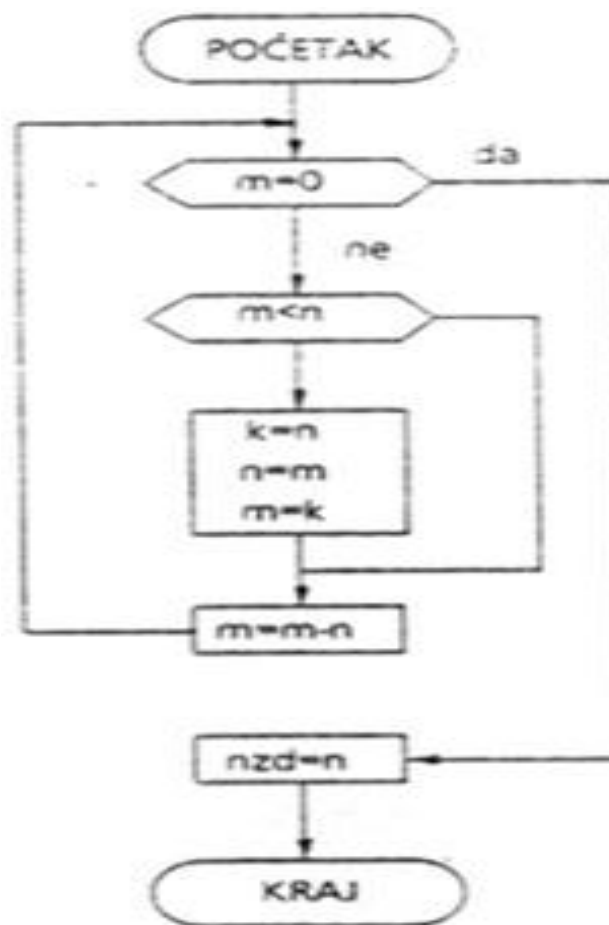
```
    return x;
```

Zapis algoritma - *NZD*

- Algoritamske šeme (*dijagram toka*):

Verzija sa
oduzimanjem:

$$nzd(n,m) = ?$$



Zapis algoritma - **NZD**

- Drugo rešenje – pseudo-kod:

```
// Ulaz: pozitivni celi brojevi x i y  
// Izlaz: nzd(x,y)  
algorithm gcd(x, y)  
  
    d = min{x,y};  
    while ((x % d != 0) || (y % d != 0)) do  
        d = d - 1;  
  
    return d;
```

Algoritmi

- **Glavne teme** kojima ćemo se baviti u toku kursa:
- **Dizajn algoritama**
 - Kreativan **proces pisanja algoritma** kojim se dolazi do nedvosmislenih i preciznih instrukcija od kojih se sastoji algoritam
 - Ne postoji čarobna formula
 - Postoje **standardni metodi** (šeme) koje možemo koristiti (algoritamske paradigme)
- **Analiza algoritama**
 - Dokaz **ispravnosti**
 - Analiza **efikasnosti**

Algoritmi

- Još jedan razlog za analizu algoritama
- Uslov: dati problem mora biti **algoritamski rešiv**
- Problem koji je algoritamski rešiv ne podrazumeva jedinstveno rešenje – takvi problemi **moгу imati više rešenja** (algoritama)
- **Upoređivanje** po kvalitetu algoritama koji rešavaju isti problem

Kriterijumi kvaliteta algoritma

- **Korektnost** (*ispravnost*) (dizajn algoritma)
- **Efikasnost**
 - **Vremenska složenost**
 - Prostorna složenost
- **Jasnoća** i **jednostavnost**
- **Optimalnost**

Domaći zadatak

Napisati algoritam u pseudo kodu koji za dati niz brojeva ***a*** od ***n*** elemenata vraća indeks elementa sa najvećom vrednošću.

Ulazni parametri: ***a*** i ***n***.