

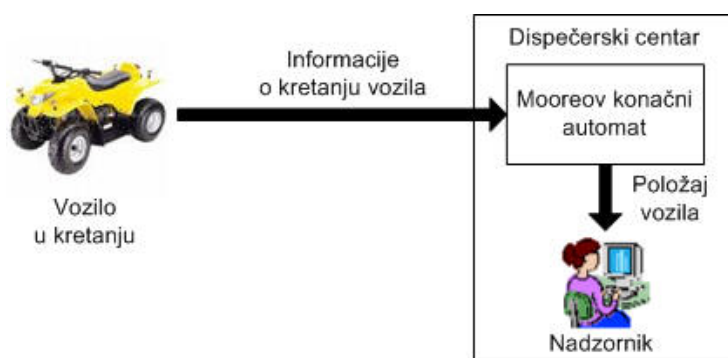
# Uvod u teoriju računarstva

## Labosi - ak. god. 2008/09

### Zadaci

#### 1. Zadatak

Zagrebačke ceste nabavile su novo vozilo za ophodnju cesta koje prometuje bez vozača. Osvremenjeni sustav za nadzor gradskih prometnica prikazan je slikom 1. Vozilo je opremljeno uređajem za bežičnu komunikaciju koji tijekom vožnje u dispečerski centar dojavljuje informacije o kretanju vozila. Dispečerskom centru dojavljuje se svako skretanje vozila i svaki trenutak kada vozilo napravi put od 100 metara. Signali koje vozilo šalje u dispečerski centar su sljedeći: **L** kada vozilo skrene lijevo, **D** kada vozilo skrene desno, **P** kada vozilo napravi put od 100 metara.



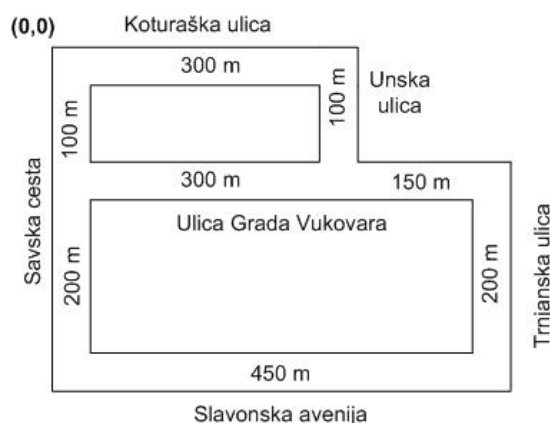
**Slika 1.** Sustav za nadzor gradskih prometnica

U dispečerskom centru podaci se obrađuju primjenom Mooreovog konačnog automata koji nadzornika sustava opskrbljuje informacijama o položaju vozila. Zahvaljujući radu Mooreovog konačnog automata, nadzornik sustava na zaslonu računala prati kretanje vozila. Područje dozvoljenog kretanja vozila zadano je kartom gradskog područja. Pretpostavlja se da su sve ulice na području dozvoljenog kretanja vozila dvosmjerne, uvijek su pravocrtne, uvijek se sijeku pod pravim kutom, mogu se pružati s obje strane raskršća, **a ukupna duljina bilo koje ulice u metrima kao i duljina bilo kojeg segmenta ulice između dvaju raskršća u metrima je višekratnik broja 50.**

#### **NADOPUNA:**

**Može se pretpostaviti da nijedna ulica na području dozvoljenog kretanja vozila nije slijepa. Također se može pretpostaviti da vozilu ni u kom trenutku nije dozvoljeno polukružno okretanje. Nailaskom na raskršće, vozilo se može nastaviti kretati ravno, može skrenuti lijevo ili desno, ali se ne može vratiti istim putem odakle je došlo. Ako vozilo u bilo kojem trenutku pokuša izvesti nedozvoljenu radnju, automat mora završiti u posebnoj stanju greške odakle se ne može oporaviti.**

Karta područja dozvoljenog kretanja vozila je parametar u radu konačnog automata i zadaje se ulaznom datotekom. Primjerice, karta gradskog područja prikazana slikom 2



**Slika 2.** Primjer karte područja dozvoljenog kretanja vozila

zadaje se ulaznom datotekom sljedećeg sadržaja:

```
Koturaška ulica,0,0,300,0
Unska ulica,300,0,300,100
Ulica Grada Vukovara,0,100,450,100
Savska cesta,0,0,0,300
Trnjanska ulica,450,300,450,100
Slavonska avenija,450,300,0,300
```

Svakim retkom ulazne datoteke definiran je položaj jedne ulice. Definicija ulice sastoji se od naziva ulice te koordinata početka i završetka ulice u metrima. Na početku rada vozilo se nalazi u gornjem lijevom kutu karte, odnosno u točki **(0,0)**, a okrenuto je prema desno. Može se pretpostaviti da barem jedna ulica počinje ili završava na tom mjestu.

### ZADATAK:

a) Programski ostvariti generator Mooreovog konačnog automata koji će u dispečerskom centru omogućiti praćenje prometovanja vozila na zadanom gradskom području. Izlaz automata mora biti pozicija vozila u jednom od dva oblika: **naziv ulice** ako vozilo prometuje ulicom pravocrtno ili **naziv raskršća** ako vozilo obavlja skretanje na raskršću dviju ili više ulica. Ako se vozilo u trenutku kada prevali put od 100 metara od posljednjeg javljanja u dispečerski centar ujedno nalazi na raskršću na kojem skreće, u dispečerski centar najprije šalje signal o prelasku puta od 100 metara, a odmah nakon toga signal o skretanju.

b) Programski ostvariti simulator Mooreovog konačnog automata koji će za zadani ulazni niz primljen od vozila u kretanju omogućiti praćenje prometovanja vozila na zadanom gradskom području.

## 2. Zadatak – Potisni automat

Konstruirati gramatiku koja generira niz naredbi pridruživanja. Za dobivenu gramatiku programski ostvariti parser tehnikom rekurzivnog spusta. Naredbe pridruživanja međusobno su odvojene znakom ";". Pojedina naredba pridruživanja ima sljedeću strukturu:

varijabla = izraz;

Lijeva strana naredbe uvijek se sastoji od jedne varijable. Desna strana naredbe može biti bilo koji aritmetički izraz koji sadrži varijable, konstante, aritmetičke operatore +, -, \* i / te otvorene i zatvorene zagrade. Na kraju svake naredbe uvijek se nalazi znak za odvajanje naredbi ";". Sve varijable zapisuju se znakom **v**, dok se sve konstante zapisuju znakom **k**. Praznine, tabulatore i znakove za novi redak parser mora ignorirati.

Nekoliko primjera ispravno zapisanog ulaznog niza:

1)  $v = k + k * v;$        $v = k + k / (v * k - (v));$

2)  $v = (((k))); v = v;$

Nekoliko primjera neispravno zapisanog ulaznog niza:

1)  $v = k + k * v;$        $v = vk + k / (v * k - (v));$

2)  $k = v + k;$

3)  $v = v + k$

Na početku rada, program treba učitati ulaznu datoteku u kojoj se nalazi zapisan ulazni niz koji se sastoji od proizvoljnog broja naredbi pridruživanja. U svakom koraku, program treba na zaslon ili u izlaznu datoteku ispisivati:

- a) znak koji trenutno učitava
- b) produkciju kojom se vrši obrada niza
- e) prihvaća li se dosad učitani dio ulaznog niza

## 3. Zadatak – Turingov stroj

Konstruirati i programski ostvariti proizvoljni model Turingovog stroja koji redom generira sve višekratnike zadanog cijelog broja. Višekratnici se generiraju redom od najmanjeg prema većima. Broj za koji je potrebno generirati višekratnike zadaje se kao parametar pri pozivu programa. Program treba omogućiti praćenje **cjelokupne konfiguracije** Turingovog stroja **u svakom koraku** rada. Program ostvariti na način da korisnik u svakom koraku može izabrati napredovanje Turingovog stroja za jedan prijelaz ili za skup prijelaza koji su potrebni da bi Turingov stroj izgenerirao novi višekratnik zadanog broja.

## Opće napomene o izvođenju laboratorijskih vježbi

- laboratorijske vježbe se obavljaju **samostalno** ili u grupama po **dva studenta** - studenti sami stvaraju parove
- studenti koji vježbu žele raditi u paru sami određuju parove unutar istog termina
- ako par studenata koji nisu u istom terminu želi zajedno raditi, onda moraju samostalno potražiti zamjenu termina s drugim studentima (dogovor o zamjeni e-mailom dojaviti voditelju vježbi)
- zamjena termina predaje moguća je samo prije prvog ciklusa laboratorijskih vježbi
- kasnije zamjene parova su moguće, ali nisu moguće zamjene termina predaje
  
- svi programi trebaju učitavati podatke iz ulaznih datoteka čija imena su zadana pri pozivu programa ili su unaprijed određena
- kôd se pregledava i **ne smije** biti komentiran
- programi mogu biti pisani u bilo kojem programskom jeziku, ali s obzirom da se predaju u fakultetskim laboratorijima trebaju raditi na jednoj od platformi koje su tamo dostupne

## Predavanje vježbi:

- vrijeme predviđeno za predaju zadataka ne predviđa i vrijeme potrebno za njihovu izradu. To znači sljedeće:
  - izradu zadatka potrebno je obaviti kod kuće
  - u terminima predaje, studenti dežurnom asistentu pokazuju i objašnjavaju svoje rješenje te demonstriraju rad programa
- svaka vježba, odnosno zadatak, predaje se zasebno
- na predaju trebaju istodobno doći oba studenta ako su vježbu radili u paru
- na predaji se provjerava poznavanje predanog programa i dijela teorije vezanog uz zadatak
  - laboratorijske vježbe nije moguće predati nakon zadanih termina

## Termini predaje:

- predaja 1. zadatka obavlja se u terminu predaje laboratorijskih vježbi prvog ciklusa u tjednu od **25.3.2008. do 28.3.2008.**
- studenti koji vježbu žele raditi u paru sami određuju parove unutar istog termina
- ako par studenata koji nisu u istom terminu želi zajedno raditi, onda moraju samostalno potražiti zamjenu termina s drugim studentima (dogovor o zamjeni e-mailom dojaviti voditelju vježbi)
- tijekom predaje vježbe, obavljat će se podjela zadataka za izradu 1. domaće zadaće (*SoftLab* zadatak)
  
- predaja 2. zadatka obavlja se u terminu predaje laboratorijskih vježbi drugog ciklusa u tjednu od **5.5.2008. do 9.5.2008.**
- tijekom predaje vježbe, dodatno će se obavljati
  - predaja 1. domaće zadaće (*SoftLab* zadatak)
  - podjela zadataka za izradu 2. domaće zadaće (samostalni studentski projekt)
- predaja 3. zadatka obavlja se u terminu predaje laboratorijskih vježbi trećeg ciklusa u tjednu od **16.6.2008. do 20.6.2008.**