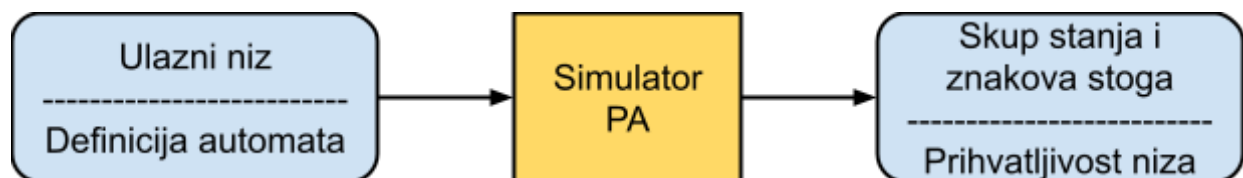


UVOD U TEORIJU RAČUNARSTVA

Ak. god. 2021/2022

3. laboratorijska vježba

U trećoj laboratorijskoj vježbi zadatak je programski ostvariti **simulator determinističkog potisnog automata (DPA)** koji nizove prihvaća prihvatljivim stanjem. Ulaz u simulator automata je tekstualni zapis njegove definicije i ulazni niz, a izlaz je tekstualni zapis skupova stanja u kojima se automat nalazio za svaki učitani znak ulaznog niza nizove znakova stoga te podatak o prihvatljivosti ulaznog niza. Rad programa koji je potrebno ostvariti u sklopu vježbe prikazan je na slici 1.



Slika 1 - Načelni rad simulatora PA

Format za zapis ulaznog niza i definicije PA je:

- 1 redak: Ulazni nizovi odvojeni znakom |. Simboli svakog pojedinog niza odvojeni su zarezom.
- 2. redak: Skup stanja odvojenih zarezom
- 3. redak: Skup ulaznih znakova odvojenih zarezom
- 4. redak: Skup znakova stoga odvojenih zarezom
- 5. redak: Skup prihvatljivih stanja odvojenih zarezom
- 6. redak: Početno stanje
- 7. redak: Početni znak stoga
- 8. redak i svi ostali retci: Funkcija prijelaza u formatu

```
trenutnoStanje, ulazniZnak, znakStoga -> novoStanje, nizZnakovaStoga
```

U skladu s definicijom u udžbeniku, znakovi stoga koji se u prijelazu stavljaju na stog stavljaju se na stog zdesna nalijevo, tj. znak na vrhu stoga nakon prijelaza je krajnje lijevi znak u nizu `nizZnakovaStoga`. Stanja, ulazni znakovi, znakovi stoga i prihvatljiva stanja bit će odvojeni zarezom (ako ih ima više od jedan). Znak praznog niza `ε` nadomješten je znakom `$`.

Automat opisan ulaznom datotekom zadovoljavat će sljedeća ograničenja:

- 1) Imena svih stanja su nizovi malih i velikih slova engleske abecede i dekadskih znamenaka. Duljina imena stanja neće biti veća od 20 znakova. Broj stanja automata neće biti veći od 20. Svako stanje automata bit će navedeno točno jednom.
- 2) Skup ulaznih znakova bit će podskup skupa malih slova engleske abecede i dekadskih znamenaka. Svaki ulazni znak će se pojaviti točno jednom. Svaki ulazni znak predstavljen je **točno jednim** znakom koji pripada malim slovima engleske abecede ili dekadskim znamenkama. Prvi redak datoteke, u kojem su ulazni nizovi, neće sadržavati više od 1500 znakova.
- 3) Skup znakova stoga bit će podskup skupa velikih slova engleske abecede. Svaki znak stoga će se pojaviti točno jednom. Svaki znak stoga predstavljen je **točno jednim** velikim slovom engleske abecede.
- 4) Skup prihvatljivih stanja će sadržavati nula ili više stanja iz skupa stanja. Svako prihvatljivo stanje bit će navedeno točno jednom.
- 5) Svaki redak funkcije prijelaza će s lijeve strane niza znakova `->` imati jedinstvenu trojku: `trenutnoStanje, ulazniZnak, znakStoga`.

Primjer ulazne datoteke s definicijom PA koji prihvaća jezik $\{w^2w^R \mid w(0+1)^*\}$ prikazan je na slici 2.

```

01      0|0,2,0|1,2,0
02      q1,q2,q3
03      0,1,2
04      J,N,K
05      q3
06      q1
07      K
08      q1,0,K->q1,NK
09      q1,1,K->q1,JK
10      q1,0,N->q1,NN
11      q1,1,N->q1,JN
12      q1,0,J->q1,NJ
13      q1,1,J->q1,JJ
14      q1,2,K->q2,K
15      q1,2,N->q2,N
16      q1,2,J->q2,J
17      q2,0,N->q2,$
18      q2,1,J->q2,$
19      q2,$,K->q3,$

```

Slika 2 - Primjer definicije PA

Ostvareni simulator PA treba ispisati u kojem se **stanju** nalazio automat za svaki ulazni znak pojedinog niza, **sadržaj stoga** te **prihvatljivost niza** (0 ili 1) nakon što je ulazni niz obrađen. Znakovi na stogu nalaze se pored pripadnog znaka stanja odvojeni znakom #, dok su parovi `stanje#znak` odvojeni znakom |. Ako se na stogu ne nalazi ništa (stog je prazan), tada se ispisuje znak \$ kao oznaka praznog stoga. Ako ulazni niz nije obrađen do kraja, a ne postoji

prijelaz potisnog automata za trenutni ulazni znak, trenutno stanje i znak na vrhu stoga, automat treba ispisati `fail` umjesto para `stanje#znak`, nakon čega automat završava s radom. Prihvatljivost niza nalazi se na kraju ispisa te je odvojena znakom `|`. Rezultati za svaki ulazni niz ispisuju se u zasebnom retku, odnosno svaki redak izlaza odvojen je znakom novog reda (`\n`). Svaki zapis započinje skupom koji sadržava početno stanje automata i početni znak stoga. Primjer izlaza za PA i ulazne nizove definirane na slici 2 prikazan je na slici 3. Brojevi s lijeve strane slike označavaju retke i nisu dio izlaza.

```
01      q1#K|q1#NK|0
02      q1#K|q1#NK|q2#NK|q2#K|q3#$|1
03      q1#K|q1#JK|q2#JK|fail|0
```

Slika 3 - Primjer izlaza simulatora automata

Važno: U slučaju prihvatanja niza (automat je pročitao cijeli ulazni niz i nalazi se u prihvatljivom stanju), zadnja konfiguracija koja se ispisuje je konfiguracija u kojoj je ostvaren uvjet prihvatljivosti, tj.| $q_n\#K_1...K_m$ |1. Ukoliko su definirani epsilon prijelazi iz prihvatljivog stanja q_n , moguće daljnje konfiguracije automata se ne ispisuju.

Napomene:

- 1) Nije potrebno provjeravati ispravnost formatiranja ulazne datoteke ili ispravnost rada automata. Neće biti preklapanja između skupa stanja i skupa simbola abecede.
- 2) Vremensko ograničenje na izvođenje programa za bilo koju ulaznu definiciju automata jest 10 sekundi
- 3) Ulazna točka za Java rješenja treba biti u razredu `SimPa`, a ulazna točka u Python rješenja treba biti u datoteci `SimPa.py`.