

# UVOD U TEORIJU RAČUNARSTVA

*Ak. God. 2015/2016*

## 4. Laboratorijska vježba

Zadatak 4. laboratorijske vježbe je programski ostvariti parser tehnikom rekurzivnog spusta za sljedeću gramatiku:

**S** → **aAB** | **bBA**

**A** → **bC** | **a**

**B** → **ccSbc** |  $\epsilon$

**C** → **AA**

Kao što je vidljivo iz produkcija, gramatika ima četiri nezavršna znaka **S**, **A**, **B** i **C**. Početni nezavršni znak je **S**. Nadalje, abeceda gramatike sastoji se od znakova **a**, **b** i **c**. Zadana gramatika je LL(1) gramatika u dodatno pojednostavljenom obliku, slično kao primjer u udžbeniku na 92. stranici.

Vašem programu će na standardni ulaz biti predan točno jedan redak koji sadrži niz znakova **a**, **b** i **c** i završava znakom za kraj retka (**\n**). Ulazni niz neće biti dulji od 200 znakova (ne računajući znak kraja retka).

Program treba parsirati niz tehnikom rekurzivnog spusta i neposredno nakon ulaska u funkciju koja obrađuje određeni nezavršni znak ispisati taj nezavršni znak na standardni izlaz. Svi nezavršni znakovi se redom zapisuju u prvi redak izlaza. Ako je niz u jeziku definiranom zadanom gramatikom, program treba u novi red (drugi redak izlaza) ispisati riječ **DA**. Ako niz nije u jeziku, u novi red treba ispisati riječ **NE**. Kada niz nije u jeziku, parser treba prestati s obradom niza što je prije moguće, tj. točno onda kada ustanovi da niz nije u jeziku. Drugim riječima, ispis je u oba slučaja jednoznačno određen. Očekivani ispis dalje je objašnjen na nekoliko primjera na kraju dokumenta.

Vremensko ograničenje na izvođenje programa za bilo koji ulazni niz jest 10 sekundi. Tipična implementacija će se za najsloženiji primjer izvesti trenutno, tj. za manje od 100 milisekundi.

Ulazna točka za Java rješenja treba biti u razredu Parser, a ulazna točka u Python rješenja treba biti u datoteci Parser.py.

Primjer 1

Ulaz:

**aa**

Očekivani ispis:

**SAB**

**DA**

Parsiranje niza **aa** započinje pozivom funkcije koja obrađuje početni nezavršni znak **S**. Kako je prvi znak niza **a**, parser jednoznačno odlučuje o primjeni produkcije **S** → **aAB**. Glava za čitanje se pomiče na sljedeći znak **a** i poziva se funkcija koja obrađuje nezavršni znak **A**. Parser jednoznačno određuje da se treba

primijeniti produkcija  $A \rightarrow a$  i funkcija koja obrađuje znak **A** pomiče glavu za čitanje na kraj niza i završava s radom. Konačno, poziva se funkcija koja obrađuje znak **B**. Kako je glava za čitanje postavljena na znak kraja niza, moguće je primijeniti jedino  $\epsilon$ -produkciju  $B \rightarrow \epsilon$ . Konačno, s obzirom na to da je pozivom funkcije koja obrađuje početni nezavršni znak **S** pročitani cijeli niz, tj. glava za čitanje se nalazi na znaku kraja niza, parser zaključuje da je niz u jeziku i u novi red ispisuje **DA**.

Primjer 2

Ulaz:

**ab**

Očekivani ispis:

**SACA**

**NE**

Isto kao u prethodnom primjeru, parser poziva funkcije za obradu znakova **S** i **A**. Pri obradi nezavršnog znaka **A**, glava za čitanje postavljena je na završni znak **b**, pa se primjenjuje produkcija  $A \rightarrow bC$ . Drugim riječima, prije poziva funkcije koja obrađuje znak **C**, glava za čitanje se pomiče na kraj niza. Funkcija koja obrađuje znak **C** ništa ne provjerava, nego odmah primjenjuje jedinu produkciju  $C \rightarrow AA$ , tj. poziva funkciju znaka **A**. Kako je glava za čitanje u tom trenutku na kraju niza, parser otkriva da niz nije u jeziku, završava obradu i ispisuje **NE**.

Primjer 3

Ulaz:

**bccaabcbbaa**

Očekivani ispis:

**SBSABACAA**

**DA**

Primjer 4

Ulaz:

**bbaab**

Očekivani ispis:

**SBACAA**

**NE**