

Snail dokumentacija

Barbara Posavec, Ivona Čižić, Luka Jandrijevic

June 2022

1 Lekser

1.1 Tipovi tokena

Lekser se sastoji od različitih simbola za usporedbu ($<$, $>$, $>=$, $<=$, $!=$, $=$...) i pridruživanje je označeno s znakom jednakosti $=$. U snailu su ključne riječi "if", "else", "then", "print", "endif", "newline" pa za njih postoje posebni tokeni. U zadaći su dodani i još dodatni tipovi poput "return", "call", "def" i "inpt" koji će služiti za vraćanje vrijednosti u funkciji, pozivanje i definiranje funkcija te "inpt" će služiti za unošenje brojeva u varijablu s tipkovnice. Također smo definirali i tokene za imena varijable (IME), za tekst te za broj. Tip podatka koji nije uobičajen u poznatim programskim jezicima (C, C++, Javascript itd...) je razlomak koji smo ovdje definirali kao token tipa BROJ.

1.2 snail lexer

Lexer smo nazvali snail. U njemu čitamo znakove i ako naiđemo na prazninu tada ju ignoriramo. Ako pročitamo znak 'j' tada ovisno o tome koji je znak sljedeći vraćamo T.MANJE ili T.MANJEJ. Slično radimo za ostale znakove. Znamo da smo naišli na neki tekst ako pročitamo navodnike te onda čitamo sve znakove dok opet ne naiđemo na navodnike koje označavaju kraj teksta/stringa. Ako pročitamo neki znak onda čitamo znakove ili brojeve te znamo da je to tada token za ime varijable/funkcije. Ako naiđemo na znak '/' i nakon toga opet na znak '/' onda znamo da je to jednolinijski komentar i zanemarujemo sav tekst u tome redu. Višelinijnski komentari su označeni znakom '#' na početku i kraju.

2 Parser

Beskontekstna gramatika je napisana u kodu. Početna varijabla je naredbe_lista pomoću koje generiramo proizvoljno varijabli imena "naredbe". "naredbe" zatim mogu biti ili pridruživanje, print, if, pozivanje funkcije, definicija funkcije ili unos varijable. Stoga, naš program se sastoji od niza takvih naredbi. Kada definiramo funkciju moramo napisati ključnu riječ "def" te zatim ime funkcije, parametre i opet listu naredbi. Na kraju definicije funkcije mora ići "return"

gdje vraćamo vrijednost izračunate funkcije. Pomoću naredbe `print` možemo ispisati ili broj/varijablu, ili tekst ili `"newline"` pomoću kojeg prelazimo u novi red. If naredba se grana s obzirom na varijablu broj te unutar ovisno o "broju" poziva se opet naredbe_lista koja sadrži listu naredbi. Varijabla "broj" može biti neka usporedba dvoje aritmetičkih izraza ili samo zasebni aritmetički izraz označen sa "aritm" koji predstavlja zbroj/razliku/množenje/djeljenje. Pri pozivu funkcija šaljemo argumente unutar zagrada i sami kontekst funkcije.

U funkciji naredbe_lista, ovisno o tome koji se token nalazi idući tada pozivamo određenu funkciju. U funkciji pridružimo pamtismo ime varijable i vrijednost koju želimo pridružiti te kreiramo određeno AST stablo. Slično u funkciji `print`, ovisno o tome što se nalazi iza tokena `T.PRINT`, pozivamo klasu `Print` s time da pamtismo što trebamo ispisati na ekran (broj, `newline` ili `string`).