

## DOKUMENTACIJA

*Programski prevodioci - predmetni zadatak*

### Osnovni podaci

Broj indeksa	IN13/2018
Ime i prezime	Djordje Stankovic
Šifra zadatka	PP-Q0Ga
Kontrolna tačka	KT2

### Evidencija implementiranog dela

		Zadatak	SIN	SEM	GK
Zajednički deo	1	Skener	+	+	+
	2	Parser	+	+	+
	3	Visestruko Deklarisanje	+	+	+
	4	Postinkrement	+	+	+
	5	Semantika za Standardne Strukture	+	+	+
	6	Visestruko Deklarisanje	+	+	+
	7	Postinkrement	+	+	+
	8	VOID	+	+	+
	9	Return	+	+	+

	10	Vise Parametara	+	+	+
	11	GK za Prethodne	+	+	+
	12	Globalne Promenljive	+	+	+
	13	Conditional Operator	+	+	+
Pojedinačni deo	1	Blokovi	+	+	+
	2	Loop	+	+	+
	3	Switch	+	+	+
Dodatni deo	1	Visestruko Definisanje sa Dodelom	+	+	+
	2	Skip i Terminate (Continue i Break)	+	+	+

## Detalji implementacije

### Z1 (Skener)

Test fajlovi: svi

Napomene:

### Z2 (Parser)

Test fajlovi: svi

Napomene:

### Z3 (Visestruko Deklarisanje)

Test fajlovi: svi

Napomene:

### Z4 (Postinkrement)

Test fajlovi: test-ok-v7-2-1.mc, test-ok-v7-2-2.mc, test-ok-v7-3-1.mc, test-ok-v7-3-2.mc, test-ok-v7-3-3.mc

Napomene: Za test-ok-kt2-v7-3-1.mc (koji je sa vezbi) sam morao da promenim return na 54 umesto 53 zbog implementacije post-inkrement-a, ja sam koristio objasnjenje na pdf-u sa vezbi, ali je trazeni rezultat dobijen preko standardnog gcc nacina inkrementiranja.

Kod mene se y++ inkrementira nakon dodele, ali u gcc-u assign to override-uje, i inkrement se nece desiti.

Na primer, za a == 1, a = a++ + a++, gcc nacin (koji ovde nije implementiran) ce raditi ovako :

- Prvo a++ ce biti 1, jer se ++ radi tek posle
- Kada dodje do + izmedju, uradio se taj ++, i sada je a postalo 2, ali je leva strana jos uvek 1
- Onda ce biti 1 + 2 sve to, a ovo drugo a++ nema uticaja jer ce ga override-ovati celokupna dodela

Da smo imali jos nesto da saberemo posle toga, onda bi se a ponovo uvecalo.

Tako da je rezultat 3, a ne 4.

Dakle ++ ce menjati sledece a, a ne to na kojem je zakaceno, osim poslednjeg a++ koji ne menja nista.

y = x++ + y++ ce po gcc-u inkrementovati x posle, ali y nece - po mojoj implementaciji ce inkrementovati i y.

Takodje, a = a++; nece da inkrementuje po gcc-u, ali ovde hoce.

Ostale testove nisam morao da menjam, jer su izgleda rezultati dobijeni ovim nacinom.

## **Z5 (Semantika za Standardne Strukture)**

Test fajlovi: svi

Napomene:

## **Z6 (Visestruko Deklarisanje)**

Test fajlovi: test-ok-NAS-33.mc

Napomene:

## **Z7 (Postinkrement)**

Test fajlovi: isto kao i kod **Z4**

Napomene:

## **Z8 (VOID)**

Test fajlovi: test-ok-NAS-81.mc ...

Napomene:

## **Z9 (Return)**

Test fajlovi: svi

Napomene: Sanity test sam promenio da ima return, jer sanity testovi ne prolaze ako baca warning.

## **Z10 (Vise Parametara)**

Test fajlovi: svi testovi koji pocinju sa test-ok-kt1-10

Napomene:

## **Z11 (GK za Prethodne)**

Test fajlovi: isto kao u **Z10**

Napomene:

## **Z12 (Globalne Promenljive)**

Test fajlovi: test-ok-kt2-v7-1-1.mc, test-ok-kt2-v7-1-2.mc ...

Napomene:

## **Z13 (Conditional Operator)**

Test fajlovi: test-ok-kt2-v8-1-1.mc, test-ok-kt2-v8-1-2.mc, test-ok-kt2-v8-1-3.mc ...

Napomene:

## **P1 (Blokovi)**

Test fajlovi: test-ok-kt1-pers1.mc, test-semerr-kt1-pers1-1.mc, test-semerr-kt1-pers1-2.mc, test-synerr-kt1-pers1-1.mc

Napomene:

**P2 (Loop)**

Test fajlovi: test-ok-kt1-pers2-1.mc, test-ok-kt1-pers2-2.mc, test-ok-kt1-pers2-3.mc ...

Napomene:

**P3 (Switch)**

Test fajlovi: test-ok-kt1-pers3-1.mc, test-ok-kt1-pers3-2.mc, test-ok-kt1-pers3-3.mc,  
test-ok-kt1-pers3-4.mc, test-ok-kt1-pers3-5.mc, test-ok-kt1-pers3-6.mc,  
test-semerr-kt1-pers3-1.mc, test-semerr-kt1-pers3-2.mc, test-semerr-kt1-pers3-3.mc

Napomene:

**D1 (Visestruko Definisanje sa Dodelom)**

Test fajlovi: test-ok-kt2-dodatni-1.mc, test-semerr-kt2-dodatni-1.mc

Napomene:

**D2 (Skip i Terminate)**

Test fajlovi: test-ok-kt2-dodatni-2-1.mc, test-ok-kt2-dodatni-2-2.mc,  
test-semerr-kt2-dodatni-2.mc, test-synerr-kt2-dodatni-2.mc

Napomene: