

*Jesenji semestar, 2021/22*

*PREDMET: SE321* *OBEZBEĐENJE KVALITETA, TESTIRANJE I ODRŽAVANJE SOFTVERA*

Domaći zadatak: **06**

Ime i prezime: **Bogdan Janković**

Broj indeksa: **3920**

Datum izrade: **12.11.2021.**

**Tekst domaćeg zadatka:** Za odabranu aplikaciju u okviru prethodnih domaćih zadataka primeniti sledeće: Testirati aplikaciju korišćenjem CodeCodera tako da se prikaže primer testiranja:

1. Pokrivenost naredbi

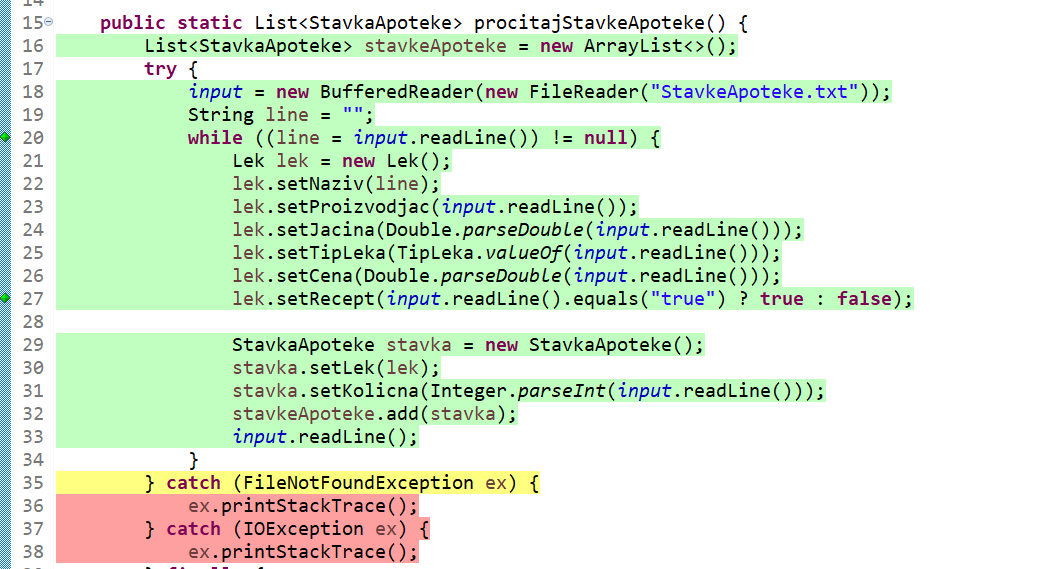
2. Pokrivenost odluka (grana)

3. Pokrivenost prostih uslova

4. Pokrivenost petlji

Za izvršeno testiranje generisati izveštaj

1. **Pokrivenost naredbi**

****

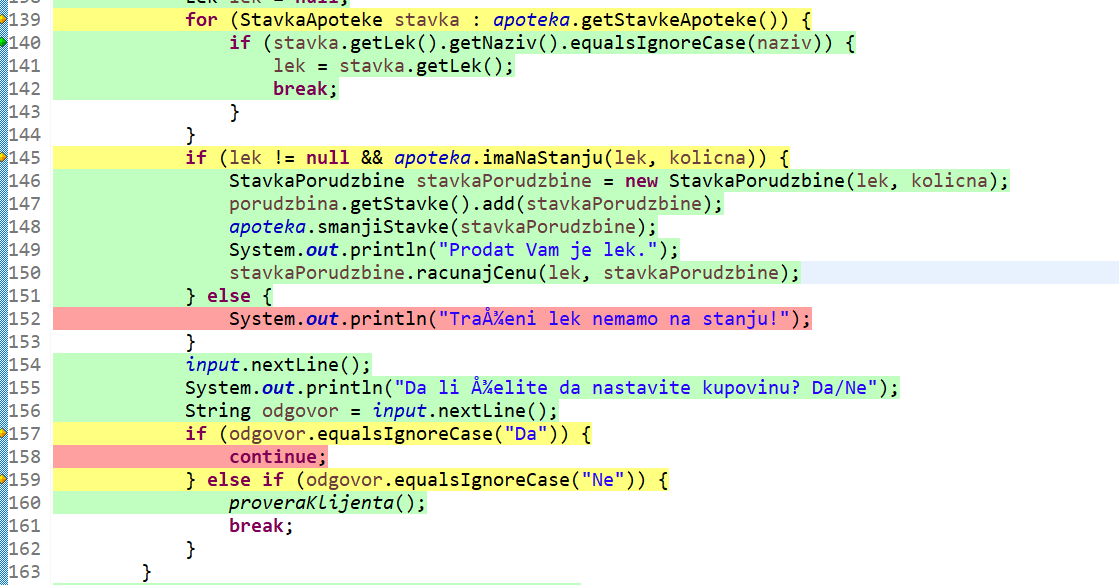
Na prikazanoj slici vidimo pokrivenost naredbi koje su neophodne za čitanje podataka, kreiranje objekata date klase i njihovo setovanje u same atribute objekata. Može videti da su sve naredbe u try bloku pokrivene ovim testom dok, kako nije bilo očekivanih izuzetaka, nije došlo ni do hvatanja odnosno pokrića istih u catch bloku.

Pri testiranju programa, test primeri se prave tako da se svaki iskaz u programu bar jednom izvrši, jer se drugačije ne može znati da li u njemu postoji greška. Međutim, ako se iskaz ispravno izvrši za jednu ulaznu vrednost, to ne znači da će se dobro izvršiti i za neku drugu ulaznu vrednost. Ovo je glavni nedostata razmatrane tehnike.

**2. Pokrivenost odluka (grana)**

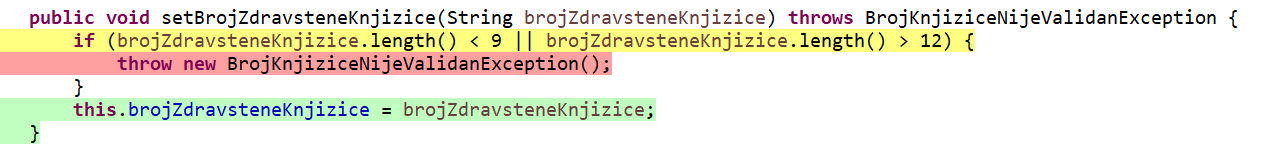
Tehnika pokrivanja odluka podrazumeva projektovanje test primera tako da se svaka od različitih grana uslovnog iskaza izvrši bar jednom. Ovo je ,jači“ metod od pokrivanja iskaza zato što pokrivanje odluka garantuje pokrivanje iskaza.

Odlučivanje se vrši na osnovu iskaza: if A then B else C Ako je uslov A ispunjen, pokrivaju se svi iskazi B koji se nalaze u odgovarajućoj grani, dok se posebno mora napisati test kada uslov A nije ispunjen, kako bi se pokrili i iskazi C.

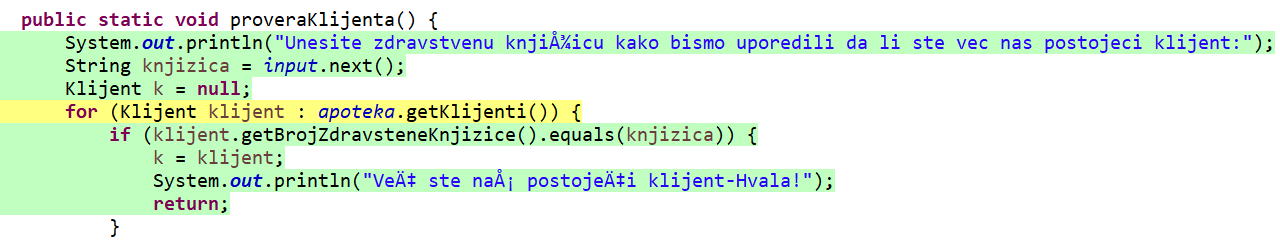


**3. Pokrivenost prostih uslova**

U tehnici pokrivanja uslova, test primeri se pišu tako da svaka elementarna komponenta nekog složenog uslova uzima vrednost iz dozvoljenog i nedozvoljenog skupa vrednosti. Elementarni uslovi se posmatraju potpuno nezavisno jedan od drugog. Ovaj metod ne garantuje pokrivanje svih odluka, pa samim tim ni pokrivanje svih iskaza u programu.



**4. Pokrivenost petlji**



Iz datog primera vidimo da je pokrivenost petlje u metodi takva da je petlja pretraživala listu klijenata sve dok nije naišla na podudaranje unetog broja knjižice sa postojećim brojem kao atributom objekta, te da nije prošla kroz celu listu odnosno izvršila sve iteracije.