

# ***Uputstva i pravila za Programiranje 1***

## TEKST ZADATAKA

- Za svaki zadatak, pored teksta zadatka, dati su i test primeri. Najčešće je dato 4 test primera, ali može biti i više.
- Dva test primera su obeležena zvezdicom i oni predstavljaju eliminacione test primere. Eliminacioni test primeri obuhvataju tipične primere izvršavanja programa koji ne obuhvataju neispravne ulazne podatke niti specijalne slučajeve.
- **Uvek se podrazumeva da sve moguće unose student treba da predvidi i da obezbedi da se program uvek ispravno ponaša.** Student treba sam da zaključi sve moguće neispravne unose i specijalne slučajeve. U test primerima se mogu uočiti neki neispravni ulazni podaci, ali neće biti istaknuti i zapisani svi.
- Ukoliko u zadatku stoji napomena "**Podrazumevati da je ulaz korektno zadatak**", onda (samo tada) nije potrebno obrađivati slučaj pogrešnog unosa podataka.

## NAČIN PREGLEDANJA I BODOVANJA C-PROGRAMA

- Program se testira automatskim pregledačem.
- Ukoliko program **nije sintaksno** ispravan dobija se **0 poena**.
- Program se testira sa **2 eliminaciona test primera (primeri sa papira obeleženi zvezdicom) i 10 neviđenih test primera** (primeri kojih nisu ponuđeni na papiru).
- Prvo se testiraju **eliminacioni test primeri** i ukoliko oba prolaze, nastavlja se sa testiranjem. Ako bar jedan eliminacioni primer ne prolazi, broj osvojenih poena 0.
- Među 10 test primera, **do 3 test primera su testovi neispravnog unosa podataka**. U testu zadatka biće naglašeno na koji izlaz treba ispisati poruku. Ukoliko se koristi pogrešan izlaz, odgovarajući test primeri neće doneti poene

iako su ispravno obrađeni. Na primer, ukoliko u tekstu zadatka piše da se greška ispisuje na standardni izlaz za greške, a u programu je greška ispisana na standardni izlaz, broj osvojenih poena za test primere neispravnog unosa je 0.

- Među 10 test primera, **do 3 test primera su specijalni slučajevi**. Specijalni slučajevi nisu pogrešan unos, ali se ponekad moraju obraditi nezavisno/odvojeno od glavnog/opšteg slučaja.
- Za svaki test primer za koji program ima ispravan izlaz dobija se 1 poen. **Testiranjem je moguće maksimalno osvojiti 10 poena.**
- Zatim se ocenjuje kvalitet koda. Kvalitetom koda se ocenjuje:
  - nazubljenost koda
  - jednostavnost rešenja
  - da li su datoteke ispravno zatvorene
  - da li su korišćeni makroi
  - da li su korišćene globalne konstantne promenljive
  - da li je memorija pravilno oslobođena
  - pravilno čuvanje zadataka

Na praktičnom ispitu je potrebno **striktno poštovati formu ispisa koja se zahteva**. Na primer, "... program ispisuje jedan ceo broj...", onda program treba da ispisuje

```
printf("%d\n", broj);
```

a **pogrešno** bi bilo ispisati:

```
printf("Resenje je: %d\n", broj);
```

Kao što se iz primera može videti, pogrešno je ispisivati bilo kakav dodatni tekst koji se prema tekstu zadatka ne traži.

Sa druge strane, dozvoljeno je ispisivati praznine, odnosno blanko, tab (\t) i znak za novi red (\n).

Ukoliko se ne ispoštuje tražena forma, pregledač vraća 0 poena. Nakon žalbi, ukoliko je jedina greška trivijalan propust tester se ponovo pokreće, a broj poena koje ispiše tester biće umanjani za 20%.

**Preporuka je da se uvek pri prevođenju koristi opcija -Wall** i da se isprave sva upozorenja koja prevodilac prijavi. Na taj način se smanjuje mogućnost grešaka, među kojima je i neinicijalizovana promenljiva.

Ukoliko je zadatak rešen tako što su **podaci učitani na pogrešan način, broj osvojenih poena je 0**. Na primer, ako se čita iz datoteke, a trebalo je sa standardnog ulaza ili ako se čita iz argumenata komandne linije, a trebalo je iz datoteke itd.

Ispit se radi na fakultetskim računarima i treba **voditi računa o inicijalizaciji promenljivih**. Na primer, može se dogoditi da se izostavi inicijalizacija (recimo  $s = 0$ ), a da se program ispravno ponaša. Ipak, tokom testiranja ova greška će biti uočena i broj osvojenih poena je 0.

Ukoliko je student ostavio **samo izvršnu verziju programa**, ali ne i kod, broj osvojenih **poena je 0**.

**Ukoliko student napravi rešenje u kome su izlazi ispravni, ali ne uradi onako kako se traži u zadatku broj osvojenih poena je 0**. Na primer, ako se u zadatku traži da se napravi funkcija koja nešto radi, a takve funkcije u kodu nema, onda je broj poena 0. Ili, ako u zadatku piše da treba rešiti korišćenjem niza, a to nije urađeno onda je broj osvojenih poena 0.

**Važno je program sačuvati na način kako piše na papiru sa zadacima** i kako je naveo dežurni asistent. Ukoliko se to ne učini može se dogoditi da se zadatak izgubi (i samim tim ne može da se oceni). Takođe, može se dogoditi da se loše sačuvan program ne testira ispravno (i da se dobije 0 poena). Ukoliko se loše sačuvani program pronađe, tester se pokreće, a broj poena koje ispiše tester biće umanjani za 20%.

Najčešće se zadaci čuvaju pod svojim rednim brojem (1.c, 2.c, itd..) u folderu čiji je naziv:

**P1\_Rok\_2019\_Tok\_ImePrezime\_BrojIndeksa\_Grupa**

Na primer, P1\_Januar1\_2019\_3\_MihailoPupin\_mr16235\_1

**mr16235** – je oznaka indeksa; mr je oznaka R-smera, 16 označava godinu upisa (2016. godina), a 235 označava broj indeksa. Ako je indeks recimo 9, godina upisa 2015, a smer teorijska matematika, onda je oznaka **mm15009**.

Oznake smerova:

- mm - Teorijska matematika i primene
- mn - Primenjena matematika
- ml - Profesor matematike i računarstva
- mr - Računarstvo i informatika
- mv - Statistika, aktuarska i finansijska matematika
- ma - Astronomija
- af - Astrofizika