

KRITERIJUM OCENJIVANJA .....	2
KAKO SE PRIPREMITI? .....	5
KOMBINACIJA 1 .....	7
KOMBINACIJA 2 .....	8
KOMBINACIJA 3 .....	9
KOMBINACIJA 4 .....	10
KOMBINACIJA 5 .....	11
KOMBINACIJA 6 .....	12
KOMBINACIJA 7 .....	13
KOMBINACIJA 8 .....	14
KOMBINACIJA 9 .....	15

# KRITERIJUM OCENJIVANJA

Na testu radite tri zadatka. Svaki zadatak se ocenjuje posebno. Ocena koja se upisuje u dnevnik je prosečna ocena ta tri zadatka.

## **Ocena 5**

Ocena 5 se dodeljuje ako je zadatak urađen potpuno ispravno, odnosno ako program radi tačno onako kako je traženo i pokriva sve predviđene uslove.

Ocena 5 može se dodeliti i ako postoji jedna manja greška, pod uslovom da:

- program je funkcionalan,
- jasno se vidi razumevanje zadatka i postupka rešavanja.

Manja greška je greška koja ne menja suštinu algoritma i ne utiče na većinu slučajeva izvršavanja programa.

## **Primeri grešaka koje i dalje mogu doneti ocenu 5:**

Kombinacija 6, zadatak 2

Program pravilno proverava temperaturu i stanje grejanja, ali u jednoj poruci postoji slovna greška u tekstu ispisa.

Kombinacija 7, zadatak 1

Program ispravno obrađuje opsege udaljenosti, ali za tačno 5 km ispisuje pogrešan vid prevoza, dok su svi ostali slučajevi tačni.

Ocena 5 se ne dodeljuje ako ne radi ključni deo zadatka (na primer petlja ili računanje proseka) ili ako postoji više od jedne greške u radu programa.

## **Ocena 4**

Ocena 4 se dodeljuje ako je zadatak uglavnom dobro urađen, ali postoji manja greška u radu programa. Pored toga, mora postojati jasno pokazano razumevanje zadatka kroz ispravan algoritam ili radni list.

Program radi u većini slučajeva, ali jedan deo uslova nije pravilno obrađen ili je pogrešno protumačen. Logika rešenja je dobra, ali realizacija nije u potpunosti tačna.

## **Primeri grešaka koje i dalje mogu doneti ocenu 4:**

Kombinacija 6, zadatak 1

Program prepoznaje destinacije „škola“ i „centar“ i ispisuje odgovarajući vid prevoza, ali za „zoo vrt“ ispisuje pogrešnu poruku. Algoritam jasno prikazuje sve predviđene slučajeve.

### Kombinacija 7, zadatak 2

Program proverava brzinu interneta i pravilno razlikuje spor i brži internet, ali pogrešno proverava da li je korisnik povezan preko Wi-Fi ili mobilne mreže. Algoritam sadrži sve tražene korake.

### Kombinacija 6, zadatak 3

Postoji petlja za unos brzina i računanje proseka, ali se poruka o dozvoljenoj brzini ne određuje pravilno u svim slučajevima. Algoritam je logičan i pokazuje razumevanje postupka.

Ocena 4 se ne dodeljuje ako ne postoji jasan algoritam ili radni list koji pokazuje razumevanje, ako zadatak sadrži veće greške u logici ili ako program ne funkcioniše u većini slučajeva.

## **Ocena 3**

Ocena 3 se dodeljuje ako postoji ispravan algoritam ili radni list koji jasno prikazuje kompletno i logično rešenje zadatka, ali kod nije u potpunosti ispravan.

Program je pokretljiv, ali:

- sadrži više grešaka,
- ne ispunjava sve zahteve iz zadatka,
- daje netačne rezultate u pojedinim ili većini slučajeva.

Razumevanje postupka rešavanja je jasno vidljivo kroz algoritam ili radni list, ali realizacija u kodu nije dovoljno tačna da bi zadatak bio u potpunosti funkcionalan.

## **Primeri grešaka koje i dalje mogu doneti ocenu 3:**

### Kombinacija 6, zadatak 3

Algoritam pravilno prikazuje unos brzina do nule, računanje proseka i proveru da li je brzina dozvoljena. Program se pokreće, ali prosek nije pravilno izračunat ili uslov za dozvoljenu brzinu nije tačno postavljen.

### Kombinacija 7, zadatak 3

Algoritam ispravno predviđa unos koraka, računanje kilometara ( $1 \text{ km} = 1300 \text{ koraka}$ ) i proveru da li je cilj ispunjen. Program radi, ali kilometri se računaju pogrešnom formulom ili se završna poruka određuje pogrešno.

### Kombinacija 7, zadatak 1

Algoritam jasno razdvaja tri opsega udaljenosti, ali su u kodu granice pogrešno postavljene, pa program daje netačne rezultate.

Ocena 3 se ne dodeljuje ako algoritam ili radni list nisu ispravni ili ako ne prikazuju kompletno rešenje zadatka.

## **Ocena 2**

Ocena 2 se dodeljuje ako algoritam ili radni list nisu potpuno ispravni i ne prikazuju jasno kompletno rešenje zadatka, a kod ne ispunjava zahteve zadatka.

Program može biti pokretljiv, ali:

- sadrži veće greške u logici,
- ne obuhvata sve tražene uslove,
- daje pogrešne rezultate u većini slučajeva.

Razumevanje zadatka je delimično, ali nije dovoljno jasno i sistematično prikazano kroz algoritam ili radni list, niti je pravilno realizovano u kodu.

### **Primeri grešaka koje i dalje mogu doneti ocenu 2:**

Kombinacija 6, zadatak 3

Algoritam ne prikazuje jasno računanje proseka ili ne predviđa prekid unosa na nulu. U kodu postoji unos brzina, ali nema pravilnog računanja proseka niti tačne provere dozvoljene brzine.

Kombinacija 7, zadatak 3

Algoritam ne sadrži računanje kilometara ili proveru da li je cilj ispunjen. Program se pokreće, ali samo sabira korake bez daljih izračunavanja.

Kombinacija 6, zadatak 1

Algoritam ne obuhvata sve destinacije ili nema jasno razrađene uslove. Program prepoznaje samo jednu destinaciju, dok ostale nisu obrađene.

Ocena 2 se ne dodeljuje ako zadatak uopšte nije započet, ako ne postoji nikakav pokušaj rešavanja, niti ako je rešenje u potpunosti pogrešno i ne pokazuje razumevanje zadatka. Sam početak bez smislenog algoritma, radnog lista ili logike rešavanja nije dovoljan za ocenu 2.

# KAKO SE PRIPREMITI?

Ako napišeš 30 zadataka samostalno test ti neće biti problem.  
Ako si samo gledao kako drugi rade test će biti težak.

## 1. Vežbati sve zadatke

Potrebno je uraditi:

- sve zadatke sa časa
- sve domaće zadatke
- sve prethodne primere za ocenu

Posebno obratiti pažnju na:

- uslove (if-else, switch)
- petlje (while, do-while, for)
- nizove
- kombinovanje više uslova
- zadatke sa brojačima i sabiranjem u petlji

Ako si neki zadatak samo „prepisao“, to znači da ga zapravo nisi vežbao.

## 2. Pisati algoritam/radni list pre koda

Pre nego što počneš da pišeš program, treba da znaš:

- šta je ulaz
- koje promenljive su potrebne
- da li je potrebna petlja
- koji je uslov
- šta treba da se ispiše

Ako ne znaš da napišeš algoritam u nekoliko jasnih koraka, teško ćeš napisati ispravan kod.

## 3. Vežbati pisanje u svesci

Iako je eitsombor.rs dozvoljen, važno je:

- znati napisati kod bez kopiranja
- voditi računa o sintaksi
- pravilno pisati zagrade i tačka-zarez

Pisanje u svesci pomaže da se uoče sopstvene greške.

#### **4. Znati gde se šta nalazi**

Pošto je eitsombor.rs dozvoljen, važno je da znaš:

- gde su zadaci sa nizovima
- gde su zadaci sa petljama
- gde su zadaci sa uslovima
- gde su primeri za ocenu

Na testu nema vremena za dugo traženje.

#### **5. Prepoznati tip zadatka**

Potrebno je da znaš da proceniš:

- da li zadatak traži rad sa nizom
- da li treba petlja do unosa 0 ili određene reči
- da li se traži maksimum, minimum ili zbir
- da li treba brojati pokušaje

Ako prepoznaš tip zadatka, lako ćeš pronaći sličan primer.

#### **6. Beležiti i ispravljati sopstvene greške**

Obratiti pažnju na česte greške:

- zaboravljen tačka-zarez
- pogrešan tip promenljive
- mešanje ReadLine i WriteLine
- razlika između = i ==
- nedostaju zagrade
- pogrešan uslov u petlji

#### **7. Pitati sve što nije jasno**

Ako postoji deo gradiva koji nije potpuno jasan, potrebno je pitati pre ocenjivanja. Nejasnoće se retko razjasne same od sebe tokom testa.

Suština pripreme je sledeća: Programiranje se ne uči čitanjem rešenja, već pisanjem zadataka.

Ako razumeš šablone zadataka i znaš da ih prilagodiš, ocenjivanje neće predstavljati problem.

# KOMBINACIJA 1

## Zadatak 1

Program treba da traži od korisnika da unese ceo broj.

Ako je broj paran, program ispisuje:

„Broj je paran.“

Ako je broj neparan, program ispisuje:

„Broj je neparan.“

## Zadatak 2

Korisnik unosi dan u nedelji i sat (0–23).

Program treba da ispiše:

„Radno vreme“ – ako je radni dan (ponedeljak–petak) i sat između 8 i 20,

„Vikend odmor“ – ako je subota ili nedelja,

„Van radnog vremena“ – u bilo kom drugom slučaju.

## Zadatak 3

Program treba da traži od korisnika da unosi cele brojeve.

Unos se ponavlja dok god korisnik ne unese nulu (0).

Kada korisnik unese 0, unos se prekida, a program treba da ispiše zbir svih unetih brojeva.

## KOMBINACIJA 2

### Zadatak 1

Program treba da traži od korisnika da unese sport (fudbal, tenis, košarka).

Na osnovu unetog sporta, program ispisuje opremu koja je potrebna:

fudbal – „kopačke“

tenis – „reket“

košarka – „patike“

Ako korisnik unese nešto drugo, ispisuje: „Sport nije u ponudi.“

### Zadatak 2

Korisnik unosi naziv paketa i osnovnu cenu.

Program proverava da li paket ima popust:

osnovni – 5%,

standardni – 10%,

premium – 15%.

Ako uneseni naziv nije u ponudi, program ispisuje: „Paket nije prepoznat.“

Na kraju ispisuje ukupnu cenu nakon popusta.

### Zadatak 3

Program treba da traži od korisnika da unosi školske ocene.

Unos se ponavlja dok god korisnik ne unese nulu (0).

Kada korisnik unese 0, unos se prekida, a program računa i prikazuje prosečnu ocenu.

## KOMBINACIJA 3

### Zadatak 1

Program treba da traži od korisnika da unese jedan karakter.

Ako je karakter znak interpunkcije (! , ? , . , ;), program ispisuje:

„Uneli ste znak interpunkcije.“

U suprotnom, program ispisuje:

„Uneli ste neki drugi simbol.“

### Zadatak 2

Korisnik unosi dan i sat (0–23).

Program treba da ispiše:

„Projekcije u toku.“ – ako je bilo koji dan i sat između 17 i 23.

„Dnevne projekcije.“ – ako je vikend (subota/nedelja) i sat između 12 i 16.

„Nema projekcija u ovom terminu.“ – u bilo kom drugom slučaju.

### Zadatak 3

Program treba da traži od korisnika da unosi iznose troškova u evrima.

Unos se ponavlja dok god korisnik ne unese nulu (0).

Kada korisnik unese 0, unos se prekida, a program treba da:

izračuna i prikaže ukupnu potrošnju u evrima,

izračuna i prikaže ukupnu potrošnju u dinarima (1 evro = 117 dinara).

## KOMBINACIJA 4

### Zadatak 1

Program treba da traži od korisnika da unese jedno slovo — a, b ili c (malo ili veliko).

Program zatim treba da ispiše poruku:

ako je uneto malo slovo a, b ili c – „Uneli ste malo slovo.“

ako je uneto veliko slovo A, B ili C – „Uneli ste veliko slovo.“

ako korisnik unese bilo koje drugo slovo ili znak – „Uneli ste pogrešan karakter.“

### Zadatak 2

Korisnik unosi u konzolu dan u nedelji i sat (0–23).

Program treba da ispiše:

„Prodavnica je otvorena.“ – ako je dan (ponedeljak–subota) i sat između 7 i 21.

„Skrraćeno radno vreme.“ – ako je nedelja i sat između 8 i 14.

„Prodavnica je zatvorena.“ – u bilo kom drugom slučaju.

### Zadatak 3

Program treba da traži od korisnika da unosi trajanje aktivnosti u satima.

Unos se ponavlja dok god korisnik ne unese nulu (0).

Kada korisnik unese 0, unos se prekida, a program treba da:

izračuna ukupno trajanje u satima,

izračuna i prikaže koliko to ukupno iznosi u minutima (1 sat = 60 minuta).

## KOMBINACIJA 5

### Zadatak 1:

Program treba da pita korisnika kojim sportom želi da se bavi.

Korisnik unosom karaktera bira:

- a – fudbal
- b – košarka
- c – tenis

Program ispisuje sport koji je izabran.

Ako korisnik unese drugi karakter, program ispisuje: „Sport nije u ponudi.“

### Zadatak 2:

Korisnik unosi nivo baterije u procentima (0–100).

Ako je nivo manji od 15, program ispisuje:

„Baterija skoro prazna.“

Ako je nivo veći ili jednak 15, proverava se da li je punjač priključen (definisano u programu).

Ako jeste, ispisuje: „Punjenje u toku.“

Ako nije, ispisuje: „Uređaj radi na bateriji.“

### Zadatak 3:

Program treba da traži od korisnika da unosi cene proizvoda (pozitivne brojeve).

Unos se ponavlja sve dok korisnik ne unese nulu (0) – tada se unos prekida.

Nakon prekida unosa program treba da:

prikaže ukupan broj unetih proizvoda,

prikaže najvišu unetu cenu,

prikaže najnižu unetu cenu.

## KOMBINACIJA 6

### Zadatak 1:

Program treba da traži od korisnika da unese destinaciju.

Na osnovu unete destinacije program ispisuje preporučeni vid prevoza:

škola – peške

zoo vrt – autobus

centar – bicikl

Ako korisnik unese destinaciju koja nije u ponudi, ispisati:

„Destinacija nije prepoznata.“

### Zadatak 2:

Korisnik unosi temperaturu u prostoriji (u °C).

Ako je temperatura manja od 18, program ispisuje: „Prostorija je hladna.“

Ako je temperatura 18 ili veća, proverava se da li je grejanje uključeno (definisano u programu).

Ako jeste, ispisuje: „Grejanje je uključeno.“

Ako nije, ispisuje: „Grejanje je isključeno.“

### Zadatak 3:

Program treba da traži od korisnika da unosi brzinu vozila (u km/h).

Unos se ponavlja dok god korisnik ne unese nulu (0).

Kada korisnik unese 0, unos se prekida, a program prikazuje prosečnu brzinu kretanja i poruku „Vozili ste dozvoljenom brzinom.“ ako je prosek manji ili jednak 80, odnosno „Vozili ste nedozvoljenom brzinom.“ ako je prosek veći od 80.

## KOMBINACIJA 7

### Zadatak 1:

Program treba da traži od korisnika da unese udaljenost do odredišta (u kilometrima).

Na osnovu unete udaljenosti program ispisuje preporučeni vid prevoza:

- ako je udaljenost manja od 2 – peške
- ako je udaljenost između 2 i 5 – bicikl
- ako je udaljenost veća od 5 – autobus

### Zadatak 2:

Korisnik unosi brzinu interneta u Mbps.

Ako je brzina manja od 10, program ispisuje: „Internet je spor.“

Ako je brzina 10 ili veća, proverava se da li je korisnik povezan preko Wi-Fi mreže (definisano u programu).

Ako jeste, ispisuje: „Povezani ste preko Wi-Fi mreže.“

Ako nije, ispisuje: „Povezani ste preko mobilne mreže.“

### Zadatak 3:

Program treba da traži od korisnika da unosi broj koraka.

Unos se ponavlja dok god korisnik ne unese nulu (0).

Kada korisnik unese 0, unos se prekida, a program prikazuje prosečan broj koraka, broj pređenih kilometara (1 km = 1300 koraka) i poruku „Cilj je ispunjen.“ ako je broj pređenih kilometara veći ili jednak 10, odnosno poruku „Cilj nije ispunjen.“ ako je broj pređenih kilometara manji od 10.

## KOMBINACIJA 8

### Zadatak 1:

Tražiti od korisnika da unese broj članova numeričkog niza.

Zatim tražiti od korisnika da unese članove niza.

Nakon unosa prvo ispisati sve brojeve manje od nule, a zatim zbir prvog i poslednjeg broja u nizu.

### Zadatak 2:

Program treba da traži od korisnika da unese trenutnu količinu proizvoda u magacinu.

Nakon unosa, program treba da proveri da li je uneta količina manja od maksimalnog kapaciteta (maksimalni kapacitet je 120).

Ako jeste ispisati koliko još proizvoda može da stane u magacin.

Ako je uneta količina jednaka maksimalnom kapacitetu ispisati poruku da je dostignut maksimalni kapacitet magacina.

Ako je uneta količina veća od maksimalnog kapaciteta ispisati poruku da je prekoračen kapacitet magacina.

### Zadatak 3:

Program treba traži od korisnika da unese pozdrav na engleskom jeziku.

Unos treba da se ponavlja sve dok korisnik ne unese tačno reč – Hello. Nakon svakog pogrešnog unosa, program treba ponovo da traži unos.

Kada korisnik unese ispravan pozdrav prekida se unos, a u konzoli se ispisuje ukupan broj pokušaja koji su bili potrebni da bi korisnik uneo ispravnu reč.

## KOMBINACIJA 9

### Zadatak 1:

Tražiti od korisnika da unese broj članova numeričkog niza.

Zatim tražiti od korisnika da unese članove niza.

Nakon unosa prvo ispisati sve brojeve sa jednocifrenim indeksom, a zatim kvadrat najvećeg broja u nizu.

### Zadatak 2:

Od korisnika se traži da unese karakter u konzolu.

Nakon toga u zavisnosti od izabranog karaktera korisnik se obaveštava o njegovom nazivu (! - uzvičnik, ? – znak pitanja, % - procenat).

Unosom karaktera koji nije u zagradi gore korisnik se obaveštava da nema podataka za taj unos.

Unos se ponavlja dok god korisnik ne unese karakter . nakon čega se program prekida a korisnik se obaveštava da je prekinuo program.

### Zadatak 3:

Program treba da traži od korisnika da unese lozinku. Ispravna lozinka je unapred definisana u programu.

Nakon svakog unosa ako je lozinka neispravna, program treba da obavesti korisnika i omogući novi unos (korisnik ima najviše tri pokušaja da unese ispravnu lozinku).

Ako je lozinka ispravna, program treba da ispiše poruku o uspešnoj prijavi i prekine izvršavanje programa.

Ako korisnik nakon tri pokušaja ne unese ispravnu lozinku program treba da ispiše poruku da je pristup odbijen i da se završi.