

PRAKTIKUM IZ PROGRAMIRANJA 1

- domaći zadatak broj 1 -

Sastaviti na simboličkom mašinskom jeziku (SMJ) za picoComputer program, kojim se vrši određena vrsta obrade nad nizom celih brojeva. Glavni program treba da:

- učita sve potrebne podatke (dužinu niza/nizova, elemente niza/nizova, itd.);
- poziva odgovarajući **potprogram** koji vrši zahtevanu obradu;
- ispisuje sve dobijene rezultate;
- ponavlja prethodna tri koraka sve dok se za dužinu niza/nizova ne unese nekorektna vrednost;

Zavisno od rednog broja problema, svaki student treba da sastavi **jedan** od programa koji su dati u prilogu ovog dokumenta. Program treba da sadrži bar **jedan potprogram** koji vrši zahtevanu obradu, ali može sadržati i dodatne potprograme.

Prilikom učitavanja niza, program treba najpre da učita dužinu niza, proveriti njenu korektnost, a zatim učita elemente niza. Ukoliko dužina niza nije korektna, korektno prekinuti izvršavanje programa.

Radi boljeg testiranja programa, odaberi nekoliko dodatnih skupova podataka sa kojima će program biti testiran. Svaki primer treba da sadrži ulazne podatke i očekivani izlaz za te podatke. Odabrane skupove podataka treba prezentovati na odbrani.

Napomene:

- Rok za predaju prvog domaćeg zadatka je **utorak, 16.11.2021.** putem kursa predmeta na **Moodle** platformi za elektronsko učenje. Tačan termin za predaju će biti naknadno definisan za sve studente. Termin će biti ograničenog vremenskog trajanja.
- Prilikom predaje domaćeg zadatka studenti će rešavati i kratak test znanja u vezi sa rešenjem domaćeg zadatka i relevantnog gradiva iz oblasti programiranja na simboličkom mašinskom jeziku (SMJ) za picoComputer koje obuhvata temu domaćeg zadatka.
- Odbrana prvog domaćeg zadatka će biti održana u **sredu, 17.11.2021.** Studenti mogu da se prijave za jedan od ponuđenih termina odbrane na Moodle platformi za elektronsko učenje.
- Domaći zadaci će biti testirani i ocenjivani korišćenjem javnih i tajnih testova.
- Domaći zadaci se rešavaju **samostalno**. Predmetni nastavnici zadržavaju pravo da nakon predaje domaćih zadataka izvrše proveru sličnosti i preduzmu odgovarajuće disciplinske mere.
- Svi drugi detalji oko predaje, ocenjivanja i odbrane domaćeg zadatka će biti blagovremeno objavljeni.
- Formula za redni broj problema **i** koji treba rešavati je sledeća (R – redni broj indeksa, G – poslednje dve cifre godine upisa):

$$i = (R + G) \bmod 8$$

- Kao rešenje domaćeg zadatka potrebno je predati sadržaj sledeće datoteke:

- dz1.pca**, koja sadrži izvorni tekst programa na SMJ za pC;

0. Napisati program koji svaki element zadatog niza jednocifrenih, pozitivnih celih brojeva zamenjuje njegovim K-tim stepenom ($0 \leq K \leq 4$). Obezbediti da se u niz učitavaju samo jednocifreni, pozitivni celi brojevi.

Program treba da:

- 1) Učita dužinu niza.
- 2) Proveri da li je dužina niza korektna i prekine program ukoliko nije.
- 3) Učita broj K. Ukoliko je unesena nekorektna vrednost, prekinuti program.
- 4) Učita niz jednocifrenih, pozitivnih celih brojeva. Ponavljati učitavanje pojedinačnog elementa dok se ne unese jednocifren, pozitivan broj.
- 5) Vršiti obradu po opisanom postupku i zamenjuje elemente polaznog niza.
- 6) Ispiše dobijeni niz nakon završetka obrade na prethodno opisani način.

Primeri

Ulaz:	Izlaz:
4 3 1 2 3 4	1 8 27 64
5 2 6 8 2 3	36 64 4 9
-3	

1. Napisati program koji određuje broj elemenata koji su manji od vrednosti maksimalnog elementa i broj elemenata koji su veći od vrednosti minimalnog elementa zadatog niza celih brojeva. Pozicije elemenata koji su jednaki minimalnom ili maksimalnom elementu ispisati na izlazu u rastućem poretku.

Program treba da:

- 1) Učita dužinu niza.
- 2) Proveri da li je dužina niza korektna i prekine program ukoliko nije.
- 3) Učita niz celih brojeva.
- 4) Odredi vrednosti minimalnog i maksimalnog elementa u nizu.
- 5) Odredi broj elemenata koji su manji od vrednosti maksimalnog elementa i broj elemenata koji su veći od vrednosti minimalnog elementa zadatog niza celih brojeva.
- 6) Ispiše u pojedinačnim redovima dobijeni rezultat nakon završetka obrade na prethodno opisani način.
- 7) U rastućem poretku na standardnom izlazu ispiše pozicije elemenata koji su jednaki minimalnom ili maksimalnom elementu.

Primeri

Ulaz:	Izlaz:
6 5 -2 1 3 14 -2	5 4 1 4 5
4 1 2 1 2	2 2 0 1 2 3
-3	

2. Napisati program koji od zadatog niza nenegativnih celih brojeva formira novi niz koji sadrži samo one elemente koji su manji od razlike maksimalnog i minimalnog elementa u nizu. Elementi u novoformiranom nizu treba da budu smešteni u obrnutom poretku u odnosu na ulazni niz.

Program treba da:

- 1) Učita dužinu niza.
- 2) Proveri da li je dužina niza korektna i prekine program ukoliko nije.
- 3) Učita niz celih brojeva.
- 4) Odredi minimalni i maksimalni element zadatog niza.
- 5) Ispiše vrednost razlike maksimalnog i minimalnog elementa u nizu.
- 6) Formira novi niz na prethodno opisani način.
- 7) Ispiše dobijeni niz nakon završetka obrade na prethodno opisani način.

Primeri

Ulaz:	Izlaz:
5 5 2 1 3 7	6 3 1 2 5
3 1 1 1	0
-3	

3. Napisati program koji od niza celih brojeva formira novi niz koji sadrži samo one elemente čiji zbir cifara predstavlja jednocifren broj. Elementi u novoformiranom nizu treba da budu smešteni u istom poretku kao u zadatom nizu. Zadati niz može sadržati samo nenegativne brojeve.

Program treba da:

- 1) Učita dužinu niza.
- 2) Proveri da li je dužina niza korektna i prekine program ukoliko nije.
- 3) Učita niz celih brojeva. Ponavljati učitavanje pojedinačnog elementa dok se ne unese nenegativan broj.
- 4) Formira novi niz na prethodno opisani način.
- 5) Ispiše dobijeni niz nakon završetka obrade na prethodno opisani način.

Primeri

Ulaz:	Izlaz:
5 95 22 13 3 29	22 13 3
3 1 -5 1 1	1 1 1
-3	

4. Napisati program koji od zadatog niza celih brojeva formira novi niz čiji je svaki element jednak ostatku pri deljenju sa narednim elementom niza (sledbenikom). Smatrati da je sledbenik poslednjeg elementa niza početni element niza. Obezbediti da se u niz unose samo nenegativni brojevi.

Program treba da:

- 1) Učita dužinu niza.
- 2) Proveri da li je dužina niza korektna i prekine program ukoliko nije.
- 3) Učita niz celih brojeva. Ponavljati učitavanje pojedinačnog elementa dok se ne unese nenegativan broj.
- 4) Formira novi niz na prethodno opisani način.
- 5) Ispiše dobijeni niz nakon završetka obrade na prethodno opisani način.

Primeri

Ulaz:	Izlaz:
5 5 2 1 3 14	1 0 1 3 4
3 1 1 1	0 0 0
-3	

5. Napisati program koji ispisuje vrednosti i pozicije svih onih elemenata u zadatom nizu brojeva koji su deljivi aritmetičkom sredinom niza.

Program treba da:

- 1) Učita dužinu niza.
- 2) Proveri da li je dužina niza korektna i prekine program ukoliko nije.
- 3) Učita niz celih brojeva.
- 4) Odredi aritmetičku sredinu niza.
- 5) Ispiše aritmetičku sredinu niza.
- 6) Ispiše vrednosti i pozicije svih onih elemenata u zadatom nizu brojeva koji su deljivi aritmetičkom sredinom niza.

Primeri

Ulaz:	Izlaz:
5 2 5 1 3 14	5 5 1
3 1 1 1	1 1 0 1 1 1 2
-3	

6. Napisati program koji od dva zadata niza brojeva iste dužine formira treći, čiji je svaki element jednak vrednosti logičke ekskluzivno-ILI funkcije (XOR) nad odgovarajućim elementima prvog i drugog niza ($c[i] := a[i] \text{ XOR } b[i]$). Nizovi mogu sadržati samo nule i jedinice.

Program treba da:

- 1) Učita dužinu nizova.
- 2) Proveri da li je dužina nizova korektna i prekine program ukoliko nije.
- 3) Učita prvi niz celih brojeva.
- 4) Proveri da li se učitani niz sastoji samo od nula i jedinica. Ukoliko se ne sastoji, prekine izvršavanje programa.
- 5) Učita drugi niz celih brojeva.
- 6) Proveri da li se učitani niz sastoji samo od nula i jedinica. Ukoliko se ne sastoji, prekine izvršavanje programa.
- 7) Formira novi niz na prethodno opisani način.
- 8) Ispiše dobijeni niz nakon završetka obrade na prethodno opisani način.

Primeri

Ulaz:	Izlaz:
5 1 0 0 1 1 1 1 0 0 0	0 1 0 1 1
3 1 1 1 0 0 0	1 1 1
-3	

7. Napisati program koji od zadatog niza celih brojeva formira novi niz koji sadrži samo neparne negativne elemente i parne pozitivne elemente. Elementi u novoformiranom nizu treba da budu smešteni u obrnutom poretku u odnosu na ulazni niz.

Program treba da:

- 1) Učita dužinu niza.
- 2) Proveri da li je dužina niza korektna i prekine program ukoliko nije.
- 3) Učita niz celih brojeva.
- 4) Formira novi niz na prethodno opisani način.
- 5) Ispiše dobijeni niz nakon završetka obrade na prethodno opisani način.

Primeri

Ulaz:	Izlaz:
5 14 -2 1 -3 5	-3 14
4 16 -8 4 -2	4 16
-3	