

## 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:

- a) učita sve potrebne podatke (dužinu niza/nizova, elemente niza/nizova, itd.);
- b) poziva odgovarajući **potprogram** koji vrši zahtevanu obradu;
- c) ispisuje sve dobijene rezultate;
- d) 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, odabrati 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:

1. Rok za predaju prvog domaćeg zadatka je **utorak, 15.11.2022.** 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.
2. 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.
3. Odbrana prvog domaćeg zadatka će biti održana u **sredu, 16.11.2022.** Studenti mogu da se prijave za jedan od ponuđenih termina odbrane na Moodle platformi za elektronsko učenje.
4. Domaći zadaci će biti testirani i ocenjivani korišćenjem javnih i tajnih testova.
5. Domaći zadaci se rešavaju **samostalno**. Predmetni nastavnici zadržavaju pravo da nakon predaje domaćih zadataka izvrše proveru sličnosti i preduzmu odgovarajuće disciplinske mere.
6. Svi drugi detalji oko predaje, ocenjivanja i odbrane domaćeg zadatka će biti blagovremeno objavljeni.
7. Formula za redni broj problema **i** koji treba rešavati je sledeća (R – redni broj indeksa, G – poslednje dve cifre godine upisa):

$$i = (R + G) \bmod 4$$

8. Kao rešenje domaćeg zadatka potrebno je predati sadržaj sledeće datoteke:

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

0. Napisati program koji računa najmanji zajednički sadržalac (NZS) svakog pojedinačnog elementa niza dužine N i broja K ( $1 \leq K \leq 9$ ). Računanje NZS dva broja realizovati kao zaseban potprogram.

Program treba da:

- 1) Učita dužinu niza N, proveriti da li je učitana dužina niza korektna i prekine program ukoliko nije.
- 2) Učita broj K ( $1 \leq K \leq 9$ ), proveriti da li je učitana vrednost korektna i prekine program ukoliko nije.
- 3) Učita i ispiše elemente niza.
- 4) Za svaki element niza izračuna NZS sa brojem K i rezultat smesti u niz umesto tog elementa.
- 5) Ispiše nove elemente niza.
- 6) Ponavlja prethodne korake sve dok se unose korektne vrednosti za N i K.

Primer:

Ulaz:	Izlaz:
3	3
4	8
3 8 14	14
-1	12
	8
	28

1. Napisati program koji formira niz dužine N, čiji su elementi rezultat logičke operacije ili (OR) odrađene nad odgovarajućim elementima dva prethodno učitana niza iste dužine. Elementi svih nizova treba da budu samo brojevi 0 i 1. Računanje logičke operacije OR realizovati kao zaseban potprogram.

Program treba da:

- 1) Učita dužinu nizova N, proveriti da li je učitana dužina nizova korektna i prekine program ukoliko nije.
- 2) Učita i ispiše elemente oba niza.
- 3) Za svaki par elemenata dva niza proveriti da li imaju korektnu vrednost (0 ili 1), prekine program ukoliko nemaju, odnosno izračuna logičku operaciju OR i rezultat smesti u novoformirani niz, ukoliko imaju.
- 4) Ispiše elemente novoformiranog niza.
- 5) Ponavlja prethodne korake sve dok se unose korektne vrednosti za N i elemente nizova.

Primer:

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

2. Napisati program koji vrši konverziju svakog pojedinačnog elementa niza dužine N na sledeći način: element je potrebno konvertovati iz decimalnog oblika u binarni oblik (binarni broj interpretirati kao broj u decimalnom brojnom sistemu čije su cifre 0 i 1). Elementi niza moraju biti brojevi u opsegu [0, 31]. Konverziju broja realizovati kao zaseban potprogram.

Program treba da:

- 1) Učita dužinu niza N, proveriti da li je učitana dužina niza korektna i prekineti program ukoliko nije.
- 2) Učita i ispiše elemente niza.
- 3) Za svaki element niza proveriti da li ima korektnu vrednost, prekineti program ukoliko nema, odnosno izvrši konverziju i ispiše rezultat, ukoliko ima.
- 4) Ponavlja prethodne korake sve dok se unose korektno vrednosti za N i elemente niza.

Primer:

<b>Ulaz:</b>	<b>Izlaz:</b>
3	2
2 14 5	14
-1	5
	10
	1110
	101

3. Napisati program koji proverava da li je broj deljiv sa devet. Cifre broja se zadaju kao elementi niza dužine N (prvi element je najviša cifra, itd.). Broj je deljiv sa 9 ako je zbir njegovih cifara deljiv sa 9. Računanje zbira cifara broja realizovati kao zaseban potprogram.

Program treba da:

- 1) Učita dužinu niza N, proveriti da li je učitana dužina niza korektna i prekineti program ukoliko nije.
- 2) Redom učitava elemente niza, za svaki proveriti da li je učitana vrednost cifra, prekineti program ukoliko nije, odnosno ispiše učitani element, ukoliko jeste.
- 3) Izračuna i ispiše zbir cifara broja.
- 4) Ispiše vrednost 0 ukoliko broj nije deljiv sa 9, odnosno vrednost 1 ukoliko jeste.
- 5) Ponavlja prethodne korake sve dok se unose korektno vrednosti za N i elemente niza.

Primer:

<b>Ulaz:</b>	<b>Izlaz:</b>
3	5
5 3 7	3
-1	7
	15
	0