

Ime i Prezime: \_\_\_\_\_

JMBAG: \_\_\_\_\_

**1. Zadatak (25 bodova)**

Priručna memorija ima kapacitet 64 KB a glavna memorija je kapaciteta 4MB. Blok (linija) je veličine 512 bajtova. Priručna memorija organizirana ja kao memorija s izravnim preslikavanjem.

Odredite u koji će se bločni priključak priručne memorije preslikati blok iz glavne memorije s indeksom  $2^{11}$ .

**2. Zadatak (25 bodova)**

Slika 1. prikazuje stanje stogova za računalo na bazi MC 68000 neposredno prije izvođenja instrukcije RTE.

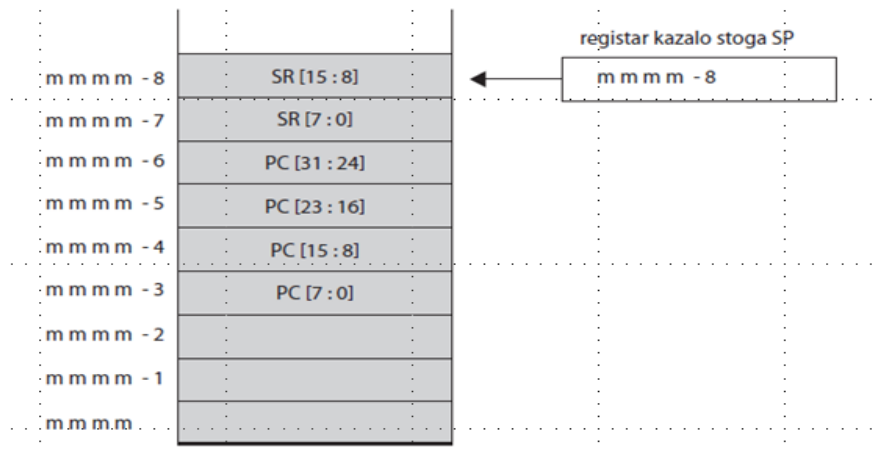
a) Odredite promijenjene sadržaje registara procesora nakon izvođenja instrukcije RTE. Odredite u kojem načinu rada je bio procesor prilikom prihvaćanja prekida. Slika 2. prikazuje organizaciju statusnog registra (SR).

b) Odredite promijenjene sadržaje registara procesora ako je programer zamijenio RTE instrukciju s programskim odsječkom

```
MOVE.W (A7)+, SR
RTS
```

(pretpostavite da se tijekom izvođenja prve instrukcije programskog odsječka nije dogodio prekid).

Ukratko komentirajte dobiveno rješenje za slučaj b).



Stanje nadglednog stoga:

mmmm-8	04
mmmm-7	06
	00
	00
	62
	00
	23
	0F

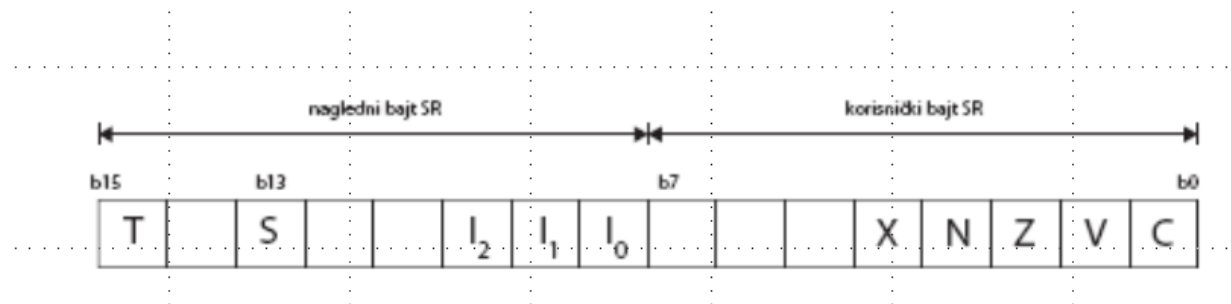
Stanje korisničkog stoga:

nnnn-8	00
nnnn-7	04
	00
	0A
	00
	00
	FF
	0F

Ime i Prezime: \_\_\_\_\_

JMBAG: \_\_\_\_\_

2. zadatak (nastavak):



sl. 2

**Ime i Prezime:** \_\_\_\_\_

**JMBAG:** \_\_\_\_\_

**3. Zadatak (25 bodova)**

Nacrtati Denningov model virtualne memorije i ukazati na namjerno ugrađenu nelogičnost u modelu.

Ukratko objasniti kako se ta "nelogičnost" rješava.

Ime i Prezime: \_\_\_\_\_

JMBAG: \_\_\_\_\_

#### 4. Zadatak (10 + 15 bodova)

Vežana lista sastoji se od čvorova sastavljenih redom od podatka (16-bitni cijeli broj) i pokazivača na sljedeći element. Kraj liste označen je pokazivačem s vrijednošću \$FFFFFFFF. Prvi čvor liste dan je na adresi LISTA.

- a) Napišite program koji pronalazi najmanji broj u listi te sprema njegov redni broj u listi na memorijsku lokaciju INDEKS.
- b) Napišite program koji **ispred** svakog čvora koji sadrži negativan broj dodaje čvor koji sadrži apsolutnu vrijednost tog broja. Nove čvorove smještajte uzastopno počevši na lokaciji PROSTOR.

**Napomene:** Ne morate deklarirati adrese LISTA, PROSTOR i INDEKS. Možete pretpostaviti da lista nije prazna.

Ime i Prezime: \_\_\_\_\_

JMBAG: \_\_\_\_\_

### 5. Zadatak (10 + 15 bodova)

- a) Napišite funkciju analognu C-ovskoj funkciji `int strlen (char* str)`, koja računa duljinu danog stringa. Kraj stringa označen je nul-znakom (nul-znak ne ulazi u duljinu). Funkcija parametre prima i vraća preko registara: u registar `D0` zapisuje duljinu stringa čiju adresu prima preko registra `A0`. Nije potrebno „paziti” na registre i stog (smiju biti promijenjeni nakon poziva).
- b) Napišite funkciju analognu `int strcat(char *s1, char *s2, int size)`. Funkcija treba zalijepiti string `s2` na kraj stringa `s1`, pod uvjetom da je duljina stringa `s2` manja ili jednaka vrijednosti `size`. Ukoliko nije, funkcija odmah izlazi.

Parametri se primaju preko stoga, tako da je `size` stavljen prvi a `s1` zadnji. Funkcija nema povratnih parametara. Potrebno je počistiti stog i vratiti originalne vrijednosti registara. Za računanje duljine stringa **obavezno** zovite funkciju iz a) dijela (čak i ako je niste napisali).

**Napomena:** Pretpostavite da su parametri već u registrima/na stogu u trenutku kada se kreću izvršavati funkcije.