

Prezime i ime (velikim slovima): _____ JMBAG: _____

Izjavljujem da tijekom izrade ove zadaće neću od drugoga primiti niti drugome pružiti pomoć, te da se neću koristiti nedopuštenim sredstvima. Ove su radnje teška povreda Kodeksa ponašanja te mogu uzrokovati i trajno isključenje s Fakulteta. Izjavljujem da mi zdravstveno stanje dozvoljava pisanje ove zadaće. Potpis: _____

Dozvoljeno je koristiti isključivo popise naredaba FRISC-a i ARM-a. Programme treba pisati uredno i komentirati pojedine cjeline programa.

Ponavljam (zaokružiti koji): 1. međuispit (1MI) 2. međuispit (2MI) Završni ispit (ZI)

1. 1MI (3 boda) Za procesor FRISC napisati potprogram X2X3 koji računa kvadrat i kub 16-bitnog broja zapisanog u formatu NBC. Potprogram prima parametar preko registra R0, a rezultati se zapisuju na adresu KVADRAT i KUB. Potprogram treba čuvati kontekst. Glavni program *ne trebate* pisati.

2. 1MI (3 boda) Za procesor FRISC napisati glavni program koji računa srednju vrijednost 8-bitnih brojeva zapisanih u formatu 2'k. Niz brojeva zapisan je u memoriji od adrese BROJEVI, a niz završava brojem FF₁₆ (on ne ulazi u računanje srednje vrijednosti). Konačan rezultat treba kao 32-bitni broj zapisati na adresu SREDNJA.

Potprogram DIJELI obavlja dijeljenje 32-bitnih brojeva zapisanih u formatu 2'k. Potprogram prima parametre za dijeljenje preko memorijskih lokacija BROJNIK i NAZIVNIK, a rezultat vraća preko registra R0. Potprogram DIJELI *ne trebate* pisati, već ga samo iskoristiti u glavnom programu.

3. 1MI (4,5 bodova) Za procesor FRISC napisati potprogram IZBACI koji iz niza 16-bitnih NBC brojeva izbacuje brojeve vrijednosti 0. Potprogram treba prepisati izvorni blok podataka u novi, izmijenjeni blok koji neće sadržavati nule. Adresu početka izvornog bloka brojeva, adresu početka izmijenjenog bloka brojeva i broj brojeva u bloku potprogram prima preko stoga. Također, potprogram treba prebrajati izbačene nule, te njihov broj vratiti preko registra R0.

Napisati glavni program koji iz memorije treba učitavati adrese početka izvornog bloka brojeva, izmijenjenog bloka brojeva i broj brojeva u bloku. Postoji ukupno 5 takvih blokova koje treba obraditi. Adrese početka izvornih blokova zapisani su u memoriji od adrese 1000₁₆, adrese početka izmijenjenih blokova zapisani su u memoriji od adrese 2000₁₆, a brojevi brojeva u bloku zapisani su u memoriji od adrese 3000₁₆. Nakon učitavanja podataka, za obradu svakog bloka potrebno je pozvati potprogram IZBACI. Primljene rezultate – broj izbačenih nula – treba spremati od adrese 4000. Potprogram treba čuvati kontekst. Parametre sa stoga treba izbrisati pozivatelj.

4. 2MI (8 bodova) Procesor FRISC upravlja mehanizmom za podizanje rampe na parkiralištu. Na sustav su spojene:

- **uvjetna** vanjska jedinica VJ1 – senzor dolaska koji očitava prisutnost i vrstu vozila pred rampom. Spremnost vanjske jedinice označava da je vozilo došlo pred rampu. Vrijednost koja se očitava sa VJ1 će biti 0F₁₆ za automobil, a F0₁₆ za kamion.
- **bezuvmjetna** vanjska jedinica VJ2 – mehanizam za podizanje/spuštanje rampe. Rampa se podiže slanjem vrijednosti 1 na VJ2, a slanje vrijednosti 0 će spustiti rampu. Ne treba brinuti o zaustavljanju procesa podizanja/spuštanja rampe
- **uvjetna** vanjska jedinica VJ3 – senzor prolaska, koji očitava da je vozilo prošlo ispod rampe. Spremnost sklopa znači da je vozilo prošlo ispod rampe, a nespremnost znači da još nije prošlo.
- sklop CT (radi u prekidnom načinu rada, spojen na INT0) – sklop koji mjeri vrijeme potrebno da rampa bude podignuta za pojedinu vrstu vozila

Napisati program koji upravlja podizanjem i spuštanjem rampe na sljedeći način. Nakon dolaska vozila pred rampu (VJ1), treba je podići slanjem vrijednosti 1 na mehanizam rampe (VJ2), te držati rampu podignutu 5 sekundi za automobil, odnosno 10 sekundi za kamion. Mjerenje vremena ostvarite pomoću sklopa CT na koji je spojen signal frekvencije 1 kHz.

Nakon isteka vremena u prekidnom potprogramu za CT treba za svaki slučaj čekati da vozilo prođe ispod rampe, provjeravanjem spremnosti senzora prolaska (VJ3) – ako je VJ3 spremna, vozilo je prošlo rampu, pa je treba spustiti slanjem vrijednosti 0 na mehanizam rampe (VJ2).

Nakon propuštanja jednog vozila, treba ponovno čekati dolazak novog vozila (to se beskonačno ponavlja).

Adrese sklopova odaberite sami.

5. **2MI, ZI (5.5 bodova)** Na procesor FRISC spojena su dva povezana CT sklopa **CT1** i **CT2** (CT2 je spojen na INTO), **PIO** (spojen na INTO) i bezuvjetna vanjska jedinica **BVJ**. Adrese vanjskih jedinica odaberite sami. Zbog jednostavnosti pretpostavite da nema prekida u prekidu, a u slučaju istovremenog prekida prvo treba poslužiti PIO pa tek onda CT2.

PIO radi u prekidnom načinu i šalje podatke procesoru, a ovaj podatke proslijeđuje na BVJ. PIO šalje **8-bitne 2'k** podatke, a na BVJ treba slati **32-bitne 2'k** podatke. Pri tome se u lokaciji PARPOZ prebraja koliko je prenesenih podataka bilo i **parno i pozitivno**. CT sklopove treba inicijalizirati tako da CT2 generira prekid na INTO svakih **20 sekundi**. Na CNT ulaz sklopa CT1 spojen je signal takta frekvencije 20MHz, a na CNT ulaz sklopa CT2 spojen je izlaz ZC od sklopa CT1. U obradi prekida za CT2 treba na memorijsku lokaciju s adresom 2000_{16} upisati koliko je podataka **preneseno na BVJ u prethodnih 20 sekundi**. Potprogrami trebaju čuvati kontekst.

Glavni program nakon inicijalizacije treba izvoditi praznu petlju.

6. **ZI (4 bodova)** Za procesor ARM napisati potprogram IZBACI koji iz niza **16-bitnih NBC brojeva izbacuje brojeve vrijednosti 0**. Potprogram treba prepisati izvorni blok podataka u novi, izmijenjeni blok koji neće sadržavati nule. Adresu početka **izvornog bloka** brojeva, adresu početka **izmijenjenog bloka** brojeva i **broj brojeva** potprogram prima preko stoga. Također, potprogram treba prebrajati izbačene nule, te njihov broj vratiti preko registra R0.

U glavnom programu treba iz memorije učitavati adrese početka izvornog bloka brojeva, izmijenjenog bloka brojeva i broj brojeva u bloku. Postoji ukupno 5 takvih blokova. Adrese početka izvornih blokova nalaze se od adrese 1000_{16} , adrese početka izmijenjenih blokova nalaze se od adrese 2000_{16} , a brojevi brojeva u bloku nalaze se od adrese 3000_{16} . Nakon učitavanja adresa svakog bloka, potrebno je za taj blok pozvati potprogram IZBACI. Primljene rezultate – broj izbačenih nula – treba spremati od adrese 4000_{16} . Potprogram treba čuvati kontekst. Parametre sa stoga treba izbrisati pozivatelj.

7. **ZI (6 bodova)** Procesor ARM upravlja **sustavom grijanja** na temelju željene temperature namještene na termostatu. Na ARM su spojeni sklopovi **GPIO1** (adresa $0xFFFF1000$) i **GPIO2** (adresa $0xFFFF2000$) te **RTC** (adresa $0xFFFF3000$, spojen na IRQ). Na vrata A sklopa GPIO1 spojen je termostat s kojeg se (na nižih 6 bitova) bezuvjetno može učitati trenutačno **namještena željena temperatura**.

Na vrata B sklopa GPIO2, na nižih 6 bitova, spojen je **termometar** kojim se očitava **stvarna trenutačna temperatura** prostorije. Svake **sekunde** treba **očitati temperaturu** i na temelju toga uključiti ili isključiti grijanje. Grijanje se uključuje ako je trenutačna temperatura **strogo manja** od željene temperature koja je trenutačno podešena na termostatu, a inače se isključuje. Grijanje se uključuje **slanjem 1** na najviši bit vrata A sklopa GPIO2, a isključuje se **slanjem 0**. Kašnjenje od jedne sekunde ostvarite sklopom RTC na čiji ulaz je spojen signal frekvencije 10 kHz.

Glavni program, nakon svih potrebnih inicijalizacija treba izvoditi beskonačnu petlju. Potprogrami trebaju čuvati kontekst.