

1. Trajanje ispita 180 minuta.
2. Odgovori se daju u vežbanci ili na formularu.
3. Na naslovnoj strani obavezno zaokružiti redne brojeve zadataka koji su rađeni.

1. KOLOKVIJUM

1. Data je rekurzivna funkcija

```
int fib(int n) {
    if (n < 2) return n;
    return fib(n-1) + fib(n-2);
}
```

$$F_n = \frac{1}{\sqrt{5}} \left(\left(\frac{1+\sqrt{5}}{2} \right)^n - \left(\frac{1-\sqrt{5}}{2} \right)^n \right)$$

$$F_n \sim \frac{1}{\sqrt{5}} \varphi^n$$

a) [9] Napisati datu funkciju korišćenjem asemblerskog jezika za RISC V procesor i koristeći standarde za prenos argumenta preko registara, čuvanja, odnosno korišćenje registara procesora. Jasno komentarisati kod. Podrazumevati da je int promenljiva dužine 32 bita.

b) [6] Da li je moguće izvesti funkciju bez posledica za bilo koju vrednost n? Koje su eventualne posledice? Ako postoje modifikovati asemblerski kod da do njih ne dolazi (nisu potrebne apsolutno tačne vrednosti nekih konstanti, ako su potrebne). Ako je za stek procesora dozvoljeno maksimalno koristiti 4KiB, koja je maksimalna vrednost za promenljivu n?

2.a) [3] Kodovati pseudoinstrukciju RISC V procesora

not x5,x6

Jasno, tabelarno, prikazati sve bite u instrukciji, kao i njihovo značenje. Na primer

.....

...	...	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
...	...	rd						OP					
...	...	0	0	1	0	1	0	0	1	0	0	1	1

b) [12] Nacrtati realizaciju dela višeciklusnog RISC V procesora koji izvodi instrukciju iz tačke a). Označiti sve signale na šemi. Jasno definisati (i objasniti šta rade) potrebne signale koje treba da generiše kontrolna jedinica, kao i njihov vremenski redosled.

3. [20] Namenski sistem koristi procesor baziran na 32bitnoj RISC-V arhitekturi instrukcijskog seta. Poznato je da je memorija povezana sa procesorom preko 32bitne magistrale koja ima odvojene putanje za podatke i za adrese. Inicijalni sadržaj dela memorije namenske platforme dat je u tabeli 3.1. Asemblerske instrukcije koje odgovaraju sadržaju memorije, date u tabeli 3.1, su predstavljene u okviru *Dissassembly 3.1*.

Dissassembly 3.1

```
0x00: addi x1, x0, 48
0x04: addi x2, x0, 52
0x08: addi x3, x0, 4
0x0C: lb x4, 0(x1)
0x10: addi x5, x4, -10
0x14: xori x5, x5, 0xA5A5
0x18: sw x5, 0(x2)
0x1C: addi x1, x1, 1
0x20: addi x2, x2, 2
0x24: addi x3, x3, -1
0x28: bne x3, x0, -28
0x2C: jal x0, 0
```

Tabela 3.1

Adresa	Sadržaj(hex)				Adresa	Sadržaj(hex)							
0x00	93	00	00	03	0x24	--	--	--	--				
0x04	13	01	40	03	0x28	--	--	--	--				
0x08	--	--	--	--	0x2C	6f	00	00	00				
0x0C	03	82	00	00	0x30	00	01	02	03				
0x10	--	--	--	--	0x34	00	00	00	00				
0x14	93	c2	52	5a	0x38	00	00	00	00				
0x18	--	--	--	--	0x3C	...							
0x1C	93	80	10	00									
0x20	13	01	21	00									

Ako su nakon sistemskog reseta svi registri procesora inicijalizovani na vrednost 0, popuniti tabele 3.2 i 3.3 (pogledati formular za odgovore) tako da predstavljaju sadržaje registara i memorijskih lokacija nakon izvršenja odgovarajuće instrukcije. Popuniti tabelu 3.4 tako da predstavlja sadržaj memorijskih lokacija čiji sadržaj nije poznat (označen sa –u tabeli 3.1).

Napomena: Sve adrese i vrednosti date u tabeli 3.1 su predstavljene u heksadecimalnom brojnom sistemu. Ukoliko ispred brojnih vrednosti **operanada** instrukcija definisanih u okviru *Dissassembly 3.1* postoji prefiks 0x smatrati da su te brojne vrednosti date u heksadecimalnom brojnom sistemu dok se u suprotnom može smatrati da su vrednosti date u decimalnom brojnom sistemu. Izvršavanje programa se analizira dok se instrukcija sa iste memorijske lokacije ne izvrši uzastopno dva puta. Broj redova u okviru tabela datih na formularima za odgovore je **proizvoljan** i ne mora odgovarati broju koraka potrebnih za potpunu analizu izvršavanja programa.

2. KOLOKVIJUM

4. [20] Vrednosti parametara hijerarhijski organizovanog memorijskog dela sistema, sa jednim stepenom hijerarhije, su:

- kapacitet glavne memorije (MC) = 256B;
- vreme pristupa glavnoj memoriji ($T_{Penalty}$) = $150T_{CLK}$;
- veličina bloka u kešu (BS) = 8B;
- adresibilna jedinica (AUS) = 1B;
- kapacitet keš memorije (CC) = 32B;
- vreme pristupa keš memoriji (T_{HIT}) = $7T_{CLK}$;
- keš memorija je organizovana kao **2way asocijativni keš** čiji kontroler primenjuje **write back – write allocate** politiku upisa;
- keš memorija je integrisana u sistem koristeći *look through* topologiju
- inicijalni sadržaj glavne memorije definisan je tabelom 3.1. Na memorijskim lokacijama čiji je sadržaj označen sa -- pretpostaviti da se nalaze vrednosti 0xAB.

Za program **P** koji se izvršava na ovoj namenskoj platformi poznato je da sekvencijalno pristupa sledećim adresama (R u indeksu označava čitanje sa memorijske lokacije definisane u uglastim zagradama dok W u indeksu označava upis podatka sa desne strane znaka = na memorijsku lokaciju u uglastim zagradama):

$M_R[0x32]$, $M_W[0x02] = 0x01$, $M_R[10]$, $M_W[0x18] = 0x02$, $M_W[0x16] = 0x03$, $M_R[0x20]$, $M_R[12]$, $M_W[0x29] = 0x04$.

a) [9] U tabeli 4.2 najpre kreirati okvir tabele koji ilustruje organizaciju keša a zatim predstaviti sadržaj kreirane tabele za svaki od adresnih ciklusa generisanih od strane procesora. U tabeli 4.1 predstaviti sadržaj dela glavne memorije nakon završetka transakcije na magistrali ukoliko dolazi do promene sadržaja memorije.

b) [3] Koliko iznosi *hit rate*? Izračunati koliko iznosi AMAT.

c) [3] Ukoliko se hijerarhijski organizovana memorija iz tačke a) proširi sa još dva dodatna stepena hijerarhije (L2 i L3) izračunati AMAT u tom slučaju ukoliko je poznato: vreme pristupa keš memoriji na L2 nivou (T_{HIT-L2}) = $7T_{CLK}$; *miss rate* na L2 nivou ($r_{MISS-L2}$) = 15%; vreme pristupa keš memoriji na L3 nivou (T_{HIT-L3}) = $75T_{CLK}$; *global miss rate* na L3 nivou ($r_{MISS-GLOBAL-L3}$) = 0.5%.

d) [5] Ukoliko je širina magistrale između procesora i keš memorije jednaka 4B, a između keš memorije i glavne memorije 8B, koliko iznosi ukupan broj generisanih transakcija na svakoj od magistrala (NT\$ - broj transakcija između procesora i keš memorije; NT\$MM – broj transakcija između keš memorije i glavne memorije)? Prodiskutovati mogućnost smanjenja ukupnog broja generisanih transakcija između keš memorije i glavne memorije.

Napomena: Ukoliko ispred brojnih vrednosti postoji prefiks 0x smatrati da su te brojne vrednosti date u heksadecimalnom brojnem sistemu dok se u suprotnom može smatrati da su vrednosti date u decimalnom brojnem sistemu. Ukoliko je potrebno izvršiti **zamenu** bloka u keš memoriji, iz keš memorije se izbacuje onaj blok koji je **prvi dodat** u keš memoriju.

5. [15] Nacrtati realizaciju memorijskog podsistema u računaru sa 32-bitnom kvazisinhronom magistralom (asinhroni pristup je kontrolisan signalom Ready sa aktivnim nivoom logičke jedinice) koja ima 20-bitnu adresnu magistralu i mogućnost upravljanja bajtovima. Veličina potrebne memorije je 128k x 8 i nalazi se u najvišem delu memorijskog prostora. Na raspolaganju su standardne memorijske komponente SRAM tipa veličine 32k x 8. Dozvoljen je samo poravnat pristup memoriji, a moguće joj je pristupiti i sa 8bitnim, 16bitnim i 32bitnim pristupom. U slučaju 16-bitnog pristupa za A1A0=00, podatak ide po delu magistrale D0..D15, a za A1A0=10 po delu magistrale D16..D31. U slučaju 8-bitnog pristupa za A1A0=00, podatak ide po delu magistrale D0..D7, za A1A0=01 po delu magistrale D8..D15, za A1A0=10 po delu magistrale D16..D23 i za A1A0=11 po delu magistrale D23..D31. Memorija nije dovoljno brza i potrebno je ubaciti čekanje u trajanju jednog taktnog ciklusa i za ciklus upisa i za ciklus čitanja.

6. [15] Nacrtati opštenamenski prekidni kontroler sa 8 ulaza koji može da prihvati prekide i sa aktivnim nivoom logičke jedinice i sa aktivnim nivoom logičke nule. Posle svakog prihvaćenog prekida, prihvaćeni prekid se automatski maskira, kao i svi prekidi nižeg nivoa. **NAPOMENA:** Procesor treba da ima mogućnost čitanja i da li postoji prekid, bez obzira da li je maskiran ili ne, i da li je sa aktivnom logičkom nulom ili jedinicom i koji prekidi su maskirani.