

OPPZURS MI 2023./24.

Zadatak 1. (2b)

Nacrtaj finite state machine brojećeg semafora.

Zadatak 2. (2b)

Koja su 3 uvjeta koja funkcija mora ispuniti kako bi bila reentrant funkcija?

Zadatak 3. (4b)

Je li za navedene situacije zadovoljen dovoljan uvjet za RMS i EDF? Zaokruži.

a) (2b)

	C_i	T_i
J_1	2	6
J_2	1	9
J_3	1	15
J_4	2	25

RMS	DA	NE
EDF	DA	NE

b) (2b)

	C_i	T_i
J_1	4	9
J_2	3	12
J_3	6	25

RMS	DA	NE
EDF	DA	NE

Zadatak 4. (2b)

Čemu služi ključna riječ `volatile` u programskom jeziku C? Napiši primjer situacije gdje je njena upotreba nužna za ispravno izvođenje programa.

Zadatak 5. (2b)

Objasni kako navedeni algoritmi raspoređuju zadatke:

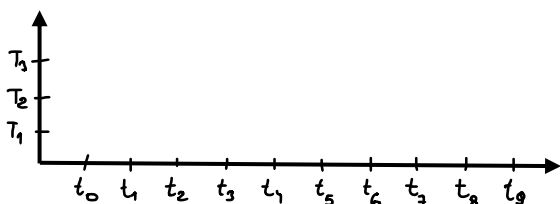
a) Rate Monotonic Scheduling (1b)

b) Earliest Deadline First (1b)

Zadatak 6. (4b)

Binarni semafor S koristi se za sinkronizaciju pristupa dijeljenom resursu od strane zadataka T1, T2, i T3. Semafor S je inicijalno slobodan. Zadatak T1 je najmanje prioritetan, dok je zadatak T3 najviše prioritetan. Zadatci su blokirani sve do pojave vanjskog događaja, kada se oni aktiviraju. Zadatak T1 čeka događaj E1, T2 čeka događaj E2, T3 čeka događaj E3. Uz zadani scenarij, nacrtaj funkciju zauzeća procesora $\sigma(t)$.

- t_0 – T1 je aktivan, T2 i T3 su blokirani
- t_1 – aktivni zadatak poziva funkciju `TakeSemaphore(S)`
- t_2 – pojava događaja E2
- t_3 – aktivni zadatak poziva funkciju `TakeSemaphore(S)`
- t_4 – pojava događaja E3
- t_5 – aktivni zadatak poziva funkciju `TakeSemaphore(S)`
- t_6 – aktivni zadatak poziva funkciju `ReleaseSemaphore(S)`
- t_7 – aktivni zadatak poziva funkciju `ReleaseSemaphore(S)`
- t_8 – aktivni zadatak poziva funkciju `ReleaseSemaphore(S)`



Zadatak 7. (4b)

Objasni postupke obrade prekida:

a) Tail-chaining (2b)

b) Late Interrupt Arrival (2b)

Zadatak 8. (4b)

Objasnite i skicirajte mehanizam sklopovske translacije iz logičkih u fizičke adrese kod korištenja virtualne memorije i MMU jedinice.

Zadatak 9. (4b)

Čemu služe registri MSP i PSP (R13)? Skicirajte vremenski dijagram na kojem će biti vidljivo koji se od registara koristi u svakom trenutku kada dolazi do prelaska iz Thread načina rada u Handler način rada, i natrag u Thread način rada.

Zadatak 10. (4b)

Nacrtaj finite state machine načina rada Cortex-M procesora.

Zadatak 11. (4b)

Bit-banding:

a) skiciraj raspored memorijskih segmenata važnih za implementaciju mehanizma i prikaži u kojoj su međusobnoj vezi. (2b)

b) napiši odsječak C programa koji postavlja vrijednost jednog bita u memoriji bez korištenja bitwise operatora. (2b)

Zadatak 12. (4b)

Nacrtaj finite state machine stanja zadatka u RTOS.