

1. Neko računalo upravlja proizvodnim procesima koji se sastoje od tri zasebne zadaće koje treba periodički obavljati (nisu poticane iz okoline već ih pokreće upravljačko računalo, tj. upravljački program koji treba osmisliti). Za prvu zadaću računalo mora unutar perioda od 100 ms napraviti obradu koja ponekad traje i do 50 ms. Druga zadaća mora biti obavljena u periodima od 500 ms, a čija obrada može potrajati i do 100 ms. Treću treba obavljati u periodima od 1 s najdužim obradama i do 200 ms.

Za zadani problem ponuditi različita rješenja (najbolja moguća ali sa slijedećim programskim strukturama, **pseudokod**) te komentirati prednosti i nedostatke:

- (2) jedna upravljačka petlja (s potrebnim strukturama podataka)
- (2) obrada zadaća u prekidnim funkcijama (uz zabranu daljnjih prekidanja do kraja pojedine obrade)
- (2) višedretveno rješenje (s prikladnim prioritetima dretvi i prioritetnim raspoređivanjem).

Pretpostaviti da u sustavu postoji podsustav za upravljanje vremenom kojim se mogu ostvariti alarmi: *postavi_alarm(T, funkcija, parametar, JEDNOM ili PERIODICKI)*; odgode izvođenja: *odgodi(koliko)*, *odgodi_do(kada)*; te dohvata trenutnog vremena: *trenutno_vrijeme(&t)*.

2. Zadan je sustav zadataka: $\tau_1: T_1=10, c_1=5$; $\tau_2: T_2=15, c_2=3$; $\tau_3: T_3=20, c_3=3$. Provjeriti rasporedivost:

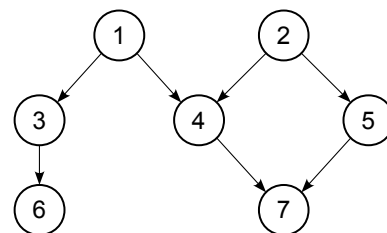
- (1) korištenjem nužnog uvjeta rasporedivosti (za RMPA raspoređivanje),
- (1) grafičkim postupkom u kritičnom slučaju (za RMPA raspoređivanje),
- (2) općim kriterijom rasporedivosti (za RMPA raspoređivanje, formula je sa strane, s time da $\lfloor X \rfloor$ predstavlja prvi veći cijeli broj od X)
- (2) ako se koristi raspoređivanje prema trenucima kranjih završetaka (DDS/EDF, prikazati rad algoritma u kritičnom trenutku do obavljanja svih zadataka)

$$(i) \quad D_i + \sum_{j=1}^{i-1} (\lfloor \frac{T_i}{T_j} \rfloor - \lfloor \frac{D_i}{T_j} \rfloor) c_j \geq T_i$$

$$(ii) \quad \sum_{j=1}^i \lfloor \frac{D_i}{T_j} \rfloor c_j \leq D_i$$

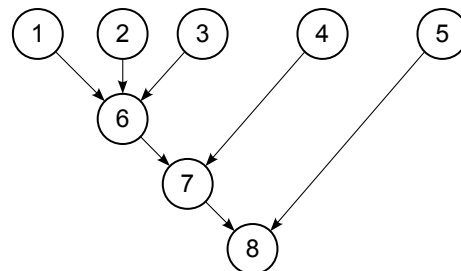
3. a) (4) Korištenjem općeg raspoređivanja (GS) napraviti raspored sustava zadataka koji je zadan necikličkim računalnim grafom. Raspoređivanje napraviti za dva procesora.

- b) (2) Prikazati mogući raspored zadataka po procesorima korištenjem rezultata općeg raspoređivanja (napraviti PS - *preemptive scheduling*).



Vremena izvođenja pojedinih zadataka (τ_1 - τ_7) su redom (c_1 - c_7): 3,5,4,2,3,6,3.

4. (3) Sustav zadataka zadan je usmjerenim grafom. Korištenjem postupka raspoređivanja sa stablenom strukturom izgraditi računalno stablo. Na stablu naznačiti visine (h). Izračunati trajanje izračunavanja cijelog sustava zadataka raspoređenog ovom metodom za slučajeve kada u sustavu postoje tri procesora. Vremena izvođenja pojedinih zadataka (τ_1 - τ_8) su redom (c_1 - c_8): 3,1,4,4,5,2,3,2.

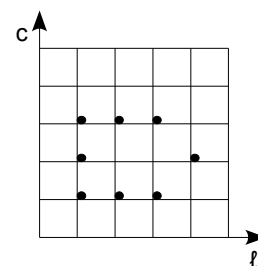


5. Sustav zadataka u trenutku t prikazan je u ℓ/c grafu.

- a) (3) Korištenjem metode krajnjih trenutaka dovršetaka zadataka (DDS/EDF) kao primarnog kriterija raspoređivanja te najmanje labavosti (LLF) kao drugog, pokazati odluke raspoređivanja do završetka svih prikazanih zadataka.

- b) (1) Izračunati zalihost procesorske snage u prve 4 jedinice vremena.

Pretpostaviti da u sustavu postoje tri procesora.



6. U nekom sustavu zbivaju se događaji koji traže obradu. Dijelovi obrada traže sredstva zaštićena binarnim semaforima. Obrada događaja obavlja se u zasebnim dretvama s odgovarajućim prioritetima (pojava događaja višeg prioriteta istiskuje obradu manjeg prioriteta, tj. odgovarajuće dretve). Neka su događaji (za promatrani vremenski interval) zadani vremenima pojave, trajanjima obrade i zahtjevima za sredstvima i njihovim oslobađanjim (na početku se zauzimaju i/ili oslobađaju sredstva):

$\tau_1: t_1=10; \{[zauzmi(s_1); radi(15)]; [zauzmi(s_2); radi(5)]; [oslobodi(s_1, s_2); radi(5)];\}$

$\tau_2: t_2=20; \{[zauzmi(s_2); radi(10)]; [zauzmi(s_1); radi(5)]; [oslobodi(s_1, s_2); radi(5)];\}$

$\tau_3: t_3=25; \{[zauzmi(s_2); radi(5)]; [zauzmi(s_1); radi(5)]; [oslobodi(s_1, s_2); radi(5)];\}$

Prikazati što će se dogoditi u sustavu ako se koristi:

- (2) samo raspoređivanje prema prioritetu
 - (2) protokol nasljeđivanja prioriteta
 - (2) protokol stropnog prioriteta – jednostavnija inačica (navesti stropne prioritete sredstava)
 - (2) protokol stropnog prioriteta – napredna inačica (navesti stropne prioritete sredstava)
7. (1) Što su to *sustavi za rad u stvarnom vremenu*? Po čemu se razlikuju od ostalih sustava?
8. (1) Koje su osnovne aktivnosti u procesu izrade programske potpore? KRATKO opišite svaku od njih.
9. (1) Opišite postupak pretvaranja ulazne (izrazite) vrijednosti u neizrazitu u sustavu upravljanja zasnovanom na neizrazitoj logici (*fuzzification*).
10. (1) Ako procesorsku iskoristivost označimo s U te broj zadataka s m , što je to $lub(U) = m(2^{1/m} - 1)$ (za sustav koji koristi RMPA te prioritetno raspoređivanje)? Što ako je $U \leq lub(U)$ a što ako je $lub(U) \leq U \leq 1$?
11. (1) Što je to rekurzivno zaključavanje i zašto je ponekad potrebno?
12. (1) Koja su uobičajena dodatna proširenja sinkronizacijskih i komunikacijskih mehanizama često potrebna u RTS okruženju (ali ne samo u njemu)?
13. (1) Koje su osnovne prepreke za korištenje običnih operacijskih sustava u RT okruženju?
14. (1) Zašto TCP nije dobar za primjenu u sustavima za rad u stvarnom vremenu? Zašto se umjesto njega preferira UDP i nad njime izgrađuju komunikacijski kanali (protokoli)?
15. (1) Kako CAN rješava problem kolizije kada dva različita uređaja istovremeno započnu sa slanjem poruke preko sabirnice?

- ① $\tau_1: T_1 = 100 \text{ } \mu\text{s} \quad c_{1\text{max}} = 50 \text{ } \mu\text{s}$
 $\tau_2: T_2 = 500 \text{ } \mu\text{s} \quad c_{2\text{max}} = 100 \text{ } \mu\text{s}$
 $\tau_3: T_3 = 1000 \text{ } \mu\text{s} \quad c_{3\text{max}} = 200 \text{ } \mu\text{s}$

a) upravljačka petlja

$T[3] = \{100, 500, 1000\}$

$t[3] = \{0, 0, 0\}$ (determinirani na početku)

ponavljaj {

 trenutno-vrijeme (&početak);

 odg = $t[1]$;

 prvi = 1;

 za $i = 2$ do 3 {

 ako ($odg > t[i]$) {

 odg = $t[i]$;

 prvi = i ;

 }

 ako ($odg > 0$)

 odgodi (odg);

 obavi-posao (prvi); $t[i] += T[i]$;

 trenutno-vrijeme (&sada);

 za $i = 1$ do 3

$t[i] -= \text{sada} - \text{početak}$;

}

b) inicijalizacija () {

 postavi-alaran (100 μs , prvi, 0, PERIODIČKI);

 —||— (200 μs , drugi, —||—);

 —||— (500 μs , treći, —||—);

}

prvi () {

 obavi što treba (obavi-posao (1) iz "a" dijela)

}

"drugi" i "treći" slično

c) dretno

inicijalizacija () {

 stvari-dretno (prva, ..., prio = 10);

 —||— (druga, ..., prio = 9);

 —||— (treća, ..., prio = 8);

}

prva () {

 ponavljaj {

 trenutno-vrijeme (&početak);

 obavi-posao (1);

 trenutno-vrijeme (& kraj);

 ako (kraj - početak < 100)

 odgodi (100 - (kraj - početak));

 } do završetka;

}

Problemi?

za a) i b): ako se dogodi da posao drugi ili treći obrade nekada bude duži (od 100 μs) prva se neće uspijeti obaviti unutar svoje periode. Dretno može pristupiti pa nema tog problema.

2. $\tau_1: \tau_1 = 10$
 $c_1 = 5$

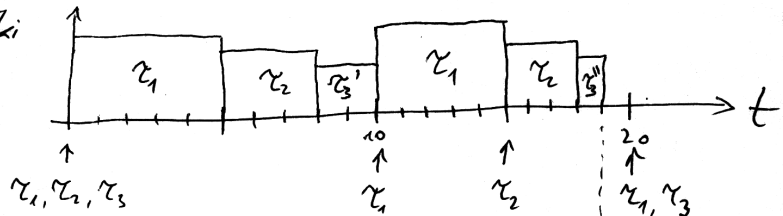
$\tau_2: \tau_2 = 15$
 $c_2 = 3$

$\tau_3: \tau_3 = 20$
 $c_3 = 3$

a) matematički $U \leq 1$

$$U = \frac{c_1}{\tau_1} + \frac{c_2}{\tau_2} + \frac{c_3}{\tau_3} = \frac{5}{10} + \frac{3}{15} + \frac{3}{20} = \frac{17}{20} \leq 1 \quad \checkmark$$

b) grafički



stiglo se sve $\checkmark$

c) opći kriterij

za τ_1 $d_i \in \{10\}$

(i) $10 \geq 10 \quad \checkmark$

(ii) $\lceil \frac{10}{10} \rceil \cdot 5 \leq 10$
 $5 \leq 10 \quad \checkmark$

za τ_2 $d_i \in \{10, 15\}$ (krenimo prvo s većim!)

$d_i = 15$ (i) $15 + (\lceil \frac{15}{10} \rceil - \lceil \frac{15}{10} \rceil) \cdot 5 \geq 15$
 $= \phi$
 $15 \geq 15 \quad \checkmark$

(ii) $\lceil \frac{15}{10} \rceil \cdot 5 + \lceil \frac{15}{15} \rceil \cdot 3 \leq 15$
 $2 \cdot 5 + 1 \cdot 3 = 13 \leq 15 \quad \checkmark$

za τ_3 $d_i \in \{10, 15, 20\}$

$d_i = 20$

(i) $20 + (\lceil \frac{20}{10} \rceil - \lceil \frac{20}{10} \rceil) \cdot 5 + (\lceil \frac{20}{15} \rceil - \lceil \frac{20}{15} \rceil) \cdot 3 \geq 20$
 $= \phi$
 $= 0$

$20 \leq 20 \quad \checkmark$

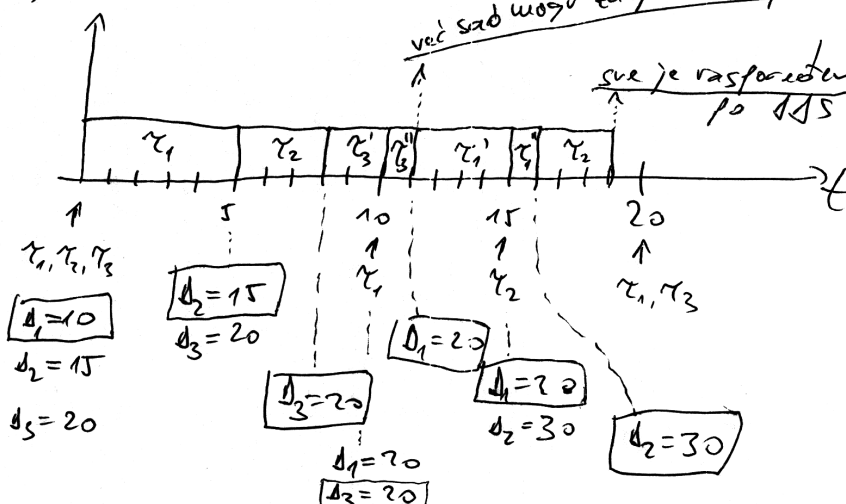
(ii) $\lceil \frac{20}{10} \rceil \cdot 5 + \lceil \frac{20}{15} \rceil \cdot 3 + \lceil \frac{20}{20} \rceil \cdot 3 \leq 20$

$2 \cdot 5 + 2 \cdot 3 + 3 = 19 \leq 20 \quad \checkmark \Rightarrow \boxed{\text{sve je raspoređeno}}$

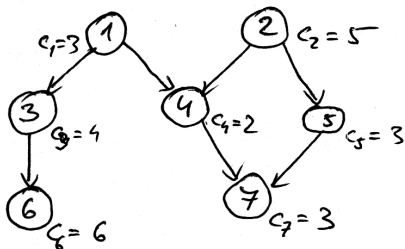
d) DDS

već sad možemo zaključiti da je sustav raspoređen!

sve je raspoređeno (u kritičnom slučaju)
 po DDS metodi



3)



a)

$$t_1 = 0$$

$$\tau_1 \quad c_1 = 3$$

$$\tau_2 \quad c_2 = 5$$

2 zad. na 2 proc.

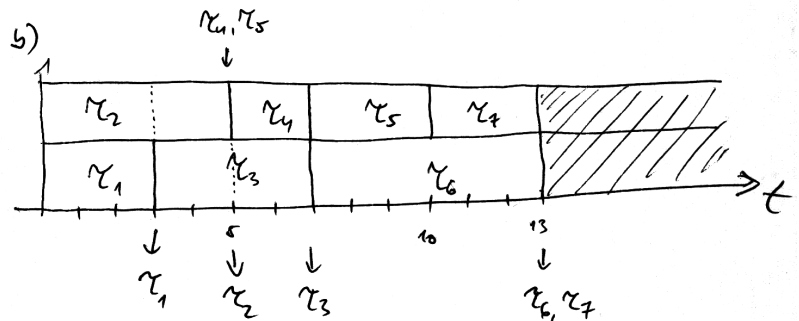
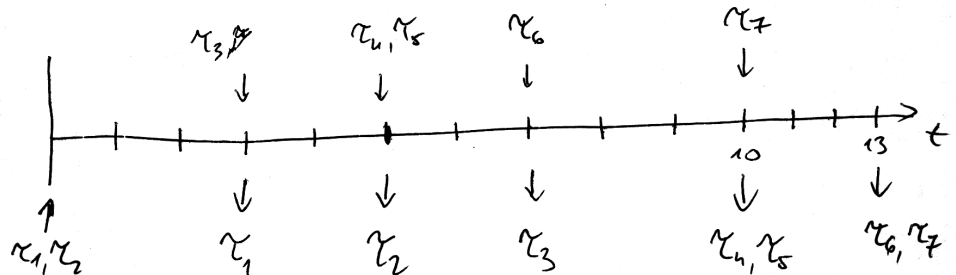
$$\alpha_1 = \alpha_2 = 1$$

$$\Delta t = \frac{c_1}{\alpha_1} = 3$$

$$t_2 = 3$$

$$\left. \begin{array}{l} \tau'_2 \quad c'_2 = 2 \\ \tau_3 \quad c_3 = 4 \end{array} \right\} \begin{array}{l} 2 \text{ zad. na 2 proc.} \\ \alpha = 1 \end{array}$$

$$\Delta t = \frac{c'_2}{\alpha} = 2$$



$$t_3 = 5$$

$$\left. \begin{array}{l} \tau'_3 \quad c'_3 = 4 - 2 = 2 \\ \tau_4 \quad c_4 = 2 \\ \tau_5 \quad c_5 = 3 \end{array} \right\} \begin{array}{l} \text{moraju se gledati} \\ \text{i ostali} \\ \text{gratni} \end{array} \quad \begin{array}{l} \tau_{\sigma_1} = \{ \tau'_3 \cup \tau_6 \} \quad c_{\sigma_1} = 8 \Rightarrow \\ \tau_{\sigma_2} = \{ \tau_4 \cup \tau_5 \cup \tau_6 \} \quad c_{\sigma_2} = 8 \end{array}$$

$$\Downarrow$$

$$\alpha_{\sigma_1} = 1$$

$$\alpha_{\sigma_2} = 1$$

$$\Delta t_1 = \frac{2}{1} = 2 \quad (\tau'_3 \text{ zavirava}, \tau_6 \text{ ulazi}) \Rightarrow t_4 = 7$$

Dakle

$$\tau_6, c_6 = 6, \alpha_6 = \alpha_{\sigma_1} = 1$$

$$\Delta t_2 = \frac{6}{1} = 6 \Rightarrow t_x = 13$$

Dakle do \tau_4 i \tau_5:

$$\alpha_4 = \frac{2}{2+3} \cdot \alpha_{\sigma_2} = \frac{2}{5} \cdot 1 = \frac{2}{5}$$

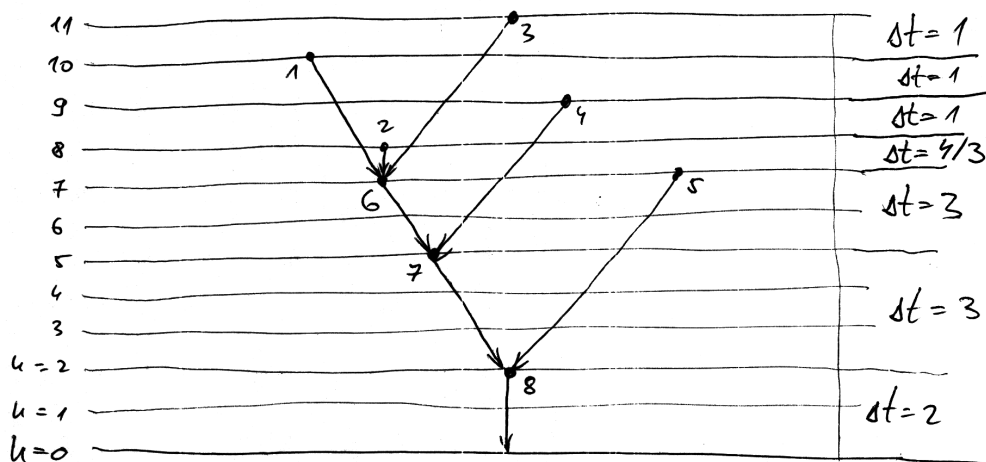
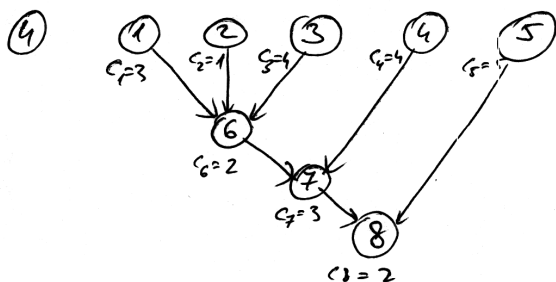
$$\alpha_5 = \frac{3}{5}$$

$$\Delta t = \frac{c_4}{\alpha_4} = \frac{2}{\frac{2}{5}} = 5 \quad \left(= \frac{c_5}{\alpha_5} = \frac{3}{\frac{3}{5}} = 5 \right)$$

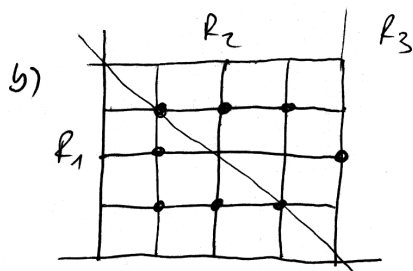
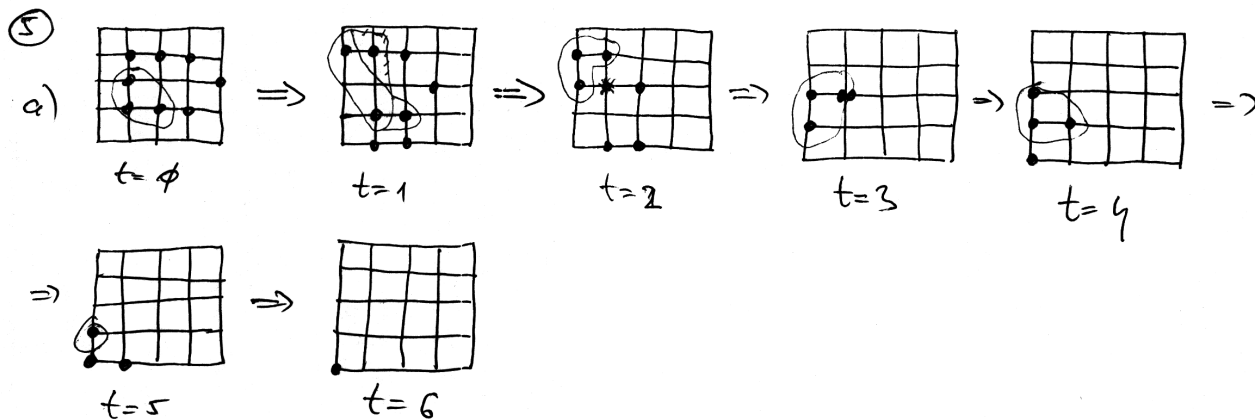
$$\Rightarrow t_7 = 10$$

$$\tau_7 \quad c_7 = 3 \quad \alpha_7 = \alpha_{\sigma_2} = 1$$

$$t_2 = 13$$



$$\Sigma = 11 \frac{1}{3}$$

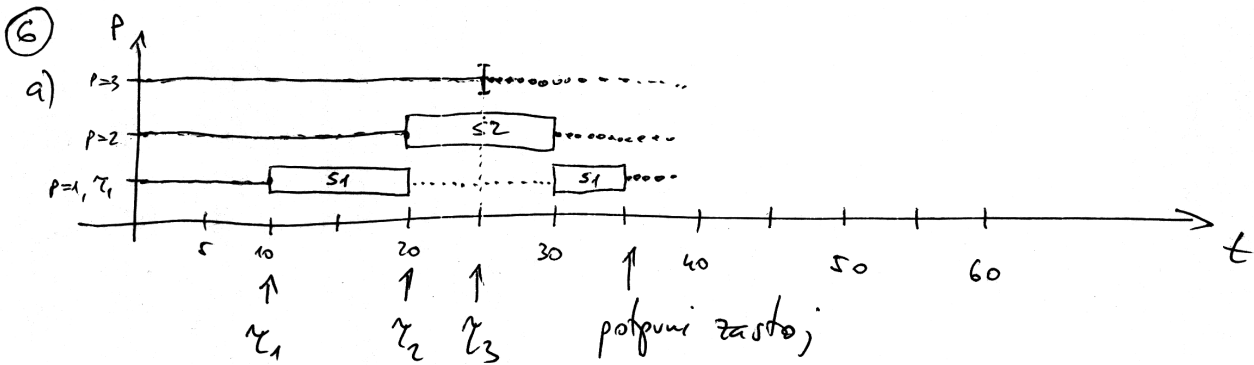


$$F(4) = 3 \cdot 4 - (1+1+1+2+3) - \{(4-2) + (4-3)\}$$

$$= 12 - 8 - 3 = 1$$

(to je vidljivo i u a) dijelu)

u $t=4$



b) isto kao i a), s time da: u $t=25$ r_2 uadijebi priustet od r_3
 u $t=30$ r_1 — " — r_2 (t. od r_3)

c) $sp(s_1) = 3$

$sp(s_2) = 3$

