

1.

a)  $t = t_m \cdot n = 1000 \mu s$

b)  $t = t_m \cdot \left( \frac{n}{4} + 3 \right) = 280 \mu s$

c)  $t = \frac{t_m \cdot n}{n} = 10 \mu s$

2. Conținutul registrelor  $r$  este interchimbabil cu celula de memorie  $m$  prin intermediul instrucțiunii  $xchg$ , ca o operație atomică.  $r_i$  este o variabilă locală a sarcinii  $S_i$  și este inițializat cu 0, iar  $m$  este o variabilă globală, comună tuturor sarcinilor, fiind inițializată cu 1.

Algoritmul pentru  $S_i$  este:

< început sarcină >;

Repetă

$xchg(r_i, m);$

până când  $r_i$ ;

< secțiune critică >;

$xchg(r_i, m);$

< sfârșit sarcină >.

Atunci când o sarcină intră în secțiune critică,  $r_i$  devine 1. Pentru excludere mutuală trebuie ca:  $\boxed{mutex(r_i) = 1}$



3.

(2)

11	8	10	3
A	B	C	D

a) codificare verticală

lungime cuvânt memorie de control este:

$$l = \log_2(11+8+10+3) = \log_2 32 = 5 \text{ bits}$$

b) codificare orizontală

$$l = 32 \text{ de bits (1 bit pt fiecare } M_i)$$

c) codificare minimală

→ clasa de compatibilitate se codifică vertical:

$$A: l_A = \log_2(11+1) = \log_2 12 = 4$$

$$B: l_B = \log_2(8+1) = \log_2 9 = 4$$

$$C: l_C = \log_2(10+1) = \log_2 11 = 4$$

$$D: l_D = \log_2(3+1) = \log_2 4 = 2$$

Într-o clasă se realizează o codificare orizontală ⇒

$$\Rightarrow l = l_A + l_B + l_C + l_D = 14 \text{ bits lungime cuvânt memorie control}$$



(3)

4. pipeline cu 4 unități paralele.

ciclul:  $F = 1$  ciclul $D = 2$  cicluri $E = 2$  cicluri $m = 3 \cdot 10^6$  instr. $D = 2 \cdot 10^9$  Hz $t = ?$  ms

$F D D E E$   
 $F D D E E$   
 $F D D E E$   
 $F D D E E$

Nr. de cicluri ( $N$ ) =  $1 + n + \frac{n}{n} \cdot c$ , unde: $n$  - nr. unități paralele (4) $c$  - cicluri de ceas pt executie (2) $m$  - nr. de instr. ( $3 \cdot 10^6$ )

$$N = 1 + 4 + \frac{3 \cdot 10^6 \cdot 2}{4} = 5 + 1500000 = 1500005 \cdot 10^6$$

$$T = \frac{\text{nr. de cicluri}}{\text{frecvență}} = \frac{N}{D} = \frac{1500005 \cdot 10^6}{2 \cdot 10^9}$$

$$T \approx 0,75 \text{ ms}$$



5.  $m_i C_1 = m_{01} m_{02} m_{03} m_{04} m_{05} m_{06}$

$m_i C_2 = m_{03} m_{07} m_{08} m_{09}$

$m_i C_3 = m_{01} m_{02} m_{08} m_{09} m_{10}$

$m_i C_4 = m_{04} m_{08} m_{11}$

$m_i C_5 = m_{06} m_{08}$

organizarea optimă?

Primum pas este să vedem dacă există  
"conflicte", adică 2 microinstrucțiuni dintr-o  
instrucțiune nu pot fi în același câmp  $\rightarrow$   
a) și b) pdece pt că în câmpul 4 se  
aflece  $m_{09}$  și  $m_{10}$  care fac parte din  $m_i C_3$ .

În c) și d) nu există "conflicte"  $\rightarrow$   
următorul pas este minimizarea costului  $\Rightarrow$  costul minim.

Se adună biții necesari pt toate câmpurile  $\rightarrow$

costul pt varianta c =  $1 + 1 + 1 + 2 + 2 + 2 + 1 = 10$

costul pt varianta d =  $1 + 1 + 1 + 1 + 2 + 2 + 1 = 9$

$\Rightarrow$  costul minim este 9  $\Rightarrow$  varianta d)



6.  $p$  procese  $\gg$   
 $N$  arbori - const.

$t_s$  - timp de sincronizare

$t_o$  - timp de overhead

care parametru influențează  
cel mai mult  $V$ ?

$t_s, t_o$  sau  $\left[\frac{N}{p}\right]$ ?

$$V = \frac{T_1}{T_p} = \frac{N \cdot t}{t_s + \left[\frac{N}{p}\right] \cdot (t + t_o)}$$

Dacă  $p \rightarrow \infty$ :  $V = \frac{N \cdot t}{t_s + \left[\frac{N}{p}\right] \cdot (t + t_o)} \rightarrow 0$

$\xrightarrow{\text{const.}}$   
 $V = \frac{N \cdot t}{t_s} \Rightarrow$  contrasă  $t_s$

Bonus: Dacă  $N \rightarrow \infty$  (ca în realitate) și  $p = \text{const}$

$$V = \frac{N \cdot t}{t_s + \left[\frac{N}{p}\right] \cdot (t + t_o)} = \frac{1}{\frac{t_s}{N \cdot t} + \frac{t + t_o}{p \cdot t}} = \frac{p \cdot t_o}{t + t_o}$$

$\xrightarrow{\text{const.}}$   
 $V = \frac{p \cdot t_o}{1 + \frac{t}{t_o}} \Rightarrow$   $t_o$  contrasă



7

???

⑥

veo idee?

8 - nu stiu daca asta a fost ordinea, dar asta  
e da meren

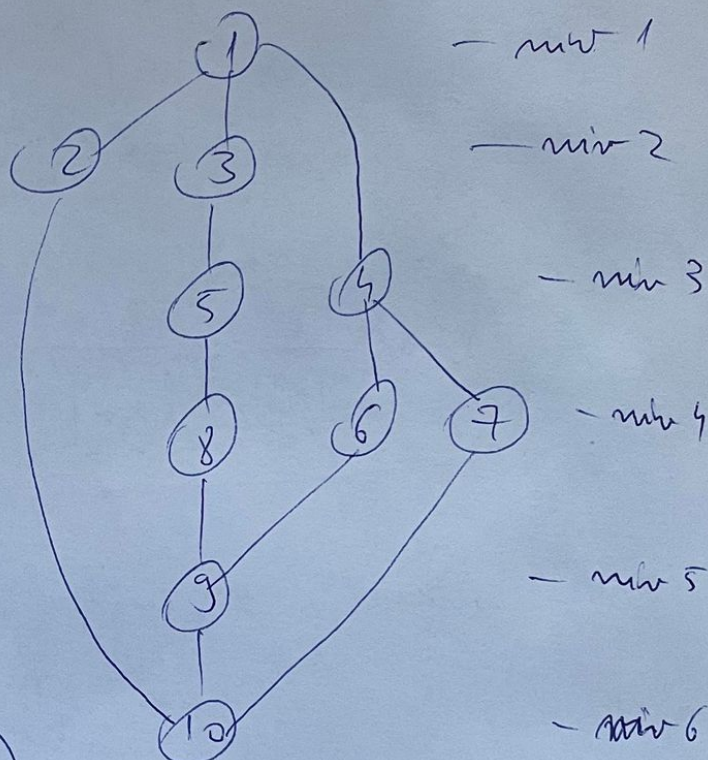
$$\mu = 480$$

$$N = 10 \text{ procese}$$

$$t = 559$$

$$f_{\text{seg}} = 6\%$$

$$V_p = ?$$



$$T_1 = f \cdot t + \frac{(1-f)t}{\mu}$$

$$= \frac{6}{100} \cdot 559 + \frac{94 \cdot 559}{100 \cdot 480}$$

② și ③ în paralel  $\Rightarrow \mu' = 480/2 = 240$  de procesoare  
 pt fiecare  $\Rightarrow T_2 = f \cdot t + \frac{(1-f)t}{\mu'} = \frac{6 \cdot 559}{100} + \frac{94 \cdot 559}{100 \cdot 240}$



4 & 5 in parallel  $\rightarrow T_3 = T_2$

(7)

6, 7 & 8 in parallel  $\rightarrow p^u = \frac{480}{3} = 160$  de  
procesare pt fiecare  $\rightarrow T_4 = \frac{6 \cdot 559}{100} + \frac{94 \cdot 559}{100 \cdot 160}$

(9)  $T_5 = T_1$

(10)  $T_6 = T_5 = T_1$

$$\begin{aligned} T_{mp} &= T_1 + T_2 + T_3 + T_4 + T_5 + T_6 \\ &= \frac{6 \cdot 559}{100} + \frac{3 \cdot 94 \cdot 559}{100 \cdot 480} + \frac{94 \cdot 559}{100 \cdot 160} + \frac{2 \cdot 94 \cdot 559}{100 \cdot 240} \\ &= \frac{36 \cdot 559}{100} + \frac{2 \cdot 94 \cdot 559}{100 \cdot 160} + \frac{2 \cdot 94 \cdot 559}{100 \cdot 240} \\ &= \frac{2 \cdot 559}{100} \left( 18 + \frac{3 \cdot 94}{160} + \frac{2 \cdot 94}{240} \right) \\ &= \frac{2 \cdot 559}{100} \cdot \frac{911}{48} \end{aligned}$$

$$V_p = \frac{T_{\text{serial}}}{T_{mp}} = \frac{N \cdot t}{T_{mp}} = \frac{10 \cdot 559 \cdot 100 \cdot 48}{2 \cdot 559 \cdot 911} \cdot 24$$

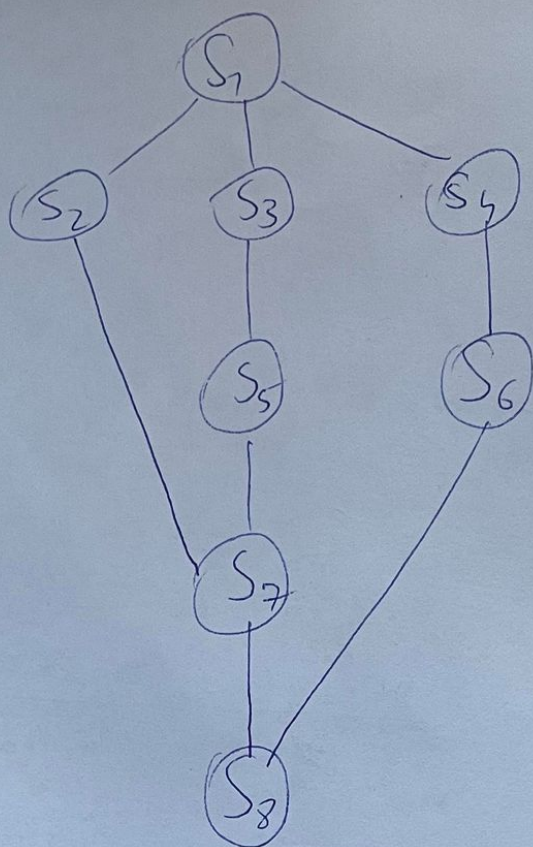
$$V_p = \frac{24000}{911} \approx \underline{\underline{26,34}}$$



9.

sistem de sarcini și dependențe:

(8)



↑ inițiere      ↑ finalizare

Oncorecurență și dependențe în SI și SF.

De ex:  $S1 \rightarrow SI_1$  și  $SF_1$ .

Vrem să ca o ~~sarcină~~ <sup>sarcină</sup> ~~recurență~~ care este dependentă de una anterioară și NU înseamnă înaintea ca "părintele" ei să fi terminat.

Varianta corectă:

$SI_1$   $SF_1$   $SI_2$   $SF_2$   $SI_3$   $SF_3$   $SI_4$   $SF_4$

$SI_5$   $SF_5$   $SI_6$   $SF_6$   $SI_7$   $SF_7$   $SI_8$   $SF_8$

10. un stiu cum se face etc. cu tabel de alcedii de sarcini... vreo idee??