

Checkers.

③ A)  $\text{Necessidade} = \text{Max} - \text{Atual}$

Need.      Max.      Attention.

P0	0	0	0	0
P1	0	7	5	0
P2	1	0	0	2
P3	0	0	2	0
P4	0	6	4	2

Available.

B) Trabalho (Processos)

Trabalho.

F0	F1	F2	F3	F4
----	----	----	----	----

1	5	2	0	7	0	0	0	0
---	---	---	---	---	---	---	---	---



+	+	+	+	+	tilibra
---	---	---	---	---	---------



→ soma com a allocation.

✓ 

1	5	3	2
---	---	---	---

 > 0 7 5 0

✓ 

2	5	3	2
---	---	---	---

 > 1 0 0 2

3	8	8	6
---	---	---	---

 > 0 0 2 0

3	14	11	8
---	----	----	---

 > 0 6 4 2

3	14	12	12
---	----	----	----

\* Ele está em estado de segurança.

→ P0, P1, P2, P3, P4.

\* São executados na ordem

→ Vai ser o novo disponível

c) (0, 4, 2, 0) feita por P1, é atendida imediatamente?

Request < Need ✓  
0 4 2 0 < 0 7 5 0

Request < Available ✓  
0 4 2 0 < 15 2 0

Process	Allocation	Need (necessidade)	Disponível
P1	1 4 2 0	0 3 3 0	1 1 0 0



B) Disponível = (1, 0, 0, 2)  $\rightarrow$  MAX - Allocation

5) Disponível = (0, 3, 0, 1)  $\rightarrow$  MAX - Allocation

	(A)				(B)			
P0	2	1	0	3	2	1	0	3
P1	1	0	0	1	1	0	0	1
P2	0	2	0	0	0	2	0	0
P3	4	1	0	2	4	1	0	2
P4	2	1	1	3	2	1	1	3

$\rightarrow$  Soma + alocação

A) 0 3 0 1 > 2 1 0 3 ✗

0 3 0 1 > 1 0 0 1 ✗

0 3 0 1 > 0 2 0 0 ✓

+ Soma com a alocação

3 4 2 2 > 4 1 0 2 ✗

$\rightarrow$  P2, P4, P0, P1, P3

3 4 2 2 > 2 1 1 3 ✓

Término

F F F F F

↓

F  $\rightarrow$  P0 ✗

F  $\rightarrow$  P1 ✗

✗  $\rightarrow$  P2

F  $\rightarrow$  P3 ✗

3 4 2 2 > 2 1 0 3 ✓  $\rightarrow$  P0

6 4 3 6 > 1 0 0 1 ✓  $\rightarrow$  P1

8 6 4 6 > 4 1 0 2 ✓  $\rightarrow$  P3

8 11 5 6

• Ordem: P2, P4, P0, P1, P3  $\rightarrow$  P4

tilibra



B)  $\begin{bmatrix} 1 & 0 & 0 & 2 \end{bmatrix} > 2103 \checkmark \rightarrow P_0$

Terminals

P <sub>0</sub>	P <sub>1</sub>	P <sub>2</sub>	P <sub>3</sub>	P <sub>4</sub>
F	F	F	F	F
+	+	+	+	+

$\begin{bmatrix} 4 & 0 & 1 & 6 \end{bmatrix} > 1001 \checkmark \rightarrow P_1$

$\begin{bmatrix} 6 & 2 & 2 & 6 \end{bmatrix} > 0200 \checkmark \rightarrow P_2$

$\begin{bmatrix} 9 & 3 & 4 & 7 \end{bmatrix} > 4102 \checkmark \rightarrow P_3$

$\begin{bmatrix} 9 & 8 & 5 & 7 \end{bmatrix} > 2113 \checkmark \rightarrow P_4$

$\begin{bmatrix} 13 & 10 & 6 & 9 \end{bmatrix}$

no Ordem: P<sub>0</sub>, P<sub>1</sub>, P<sub>2</sub>, P<sub>3</sub>, P<sub>4</sub>

A) Estado inseguro. Pode haver deadlock, ver na página anterior o seu serve.

B) Estado seguro. Atende todos os processos normalmente.

⑥ A) \* Calcular a matriz need. (max-alocação)

P <sub>0</sub>	2	2	1	1
P <sub>1</sub>	2	1	3	1
P <sub>2</sub>	0	2	1	3
P <sub>3</sub>	0	1	1	2
P <sub>4</sub>	2	2	3	3

Terminals

P <sub>0</sub>	P <sub>1</sub>	P <sub>2</sub>	P <sub>3</sub>	P <sub>4</sub>
F	F	F	F	F
+	+	+	+	+

tilibra

no Ordem dos processos: P<sub>0</sub>, P<sub>1</sub>, P<sub>2</sub>, P<sub>3</sub>, P<sub>4</sub>



Se for ✓ mesmo com a alteração. # P0, P1

3 3 2 1 > 2 2 1 1 ✓<sub>P0</sub>

5 3 3 2 > 2 1 3 1 ✓<sub>P1</sub>

8 4 5 3 > 0 2 1 3 ✓<sub>P2</sub>

10 5 5 6 > 0 1 1 2 ✓<sub>P3</sub>

11 8 6 8 > 2 2 3 3 ✓<sub>P4</sub>

12 12 9 10 \* P0, P1, P2, P3, P4

B) Solicitação (1, 1, 0, 0) → Pelo P1. Atendido imediato?

Request < need  
1 1 0 0 < 2 1 3 1 ✓

Request < <sup>Disponível</sup> available  
1 1 0 0 < 3 3 2 1 ✓

Processo	Absença	necessidade	Disponível
P1	2 0 2 1	1 0 3 1	2 2 2 1

c) (0, 0, 2, 0) → P4. → Atendido imediatamente?

0 0 2 0 < 2 2 3 3 ✓  
0 0 2 0 < 3 3 2 1 ✓

Processo	Absença	necessidade	Disponível
P4	1 4 1 2	2 2 1 3	3 3 0 1



$E = (2, 1, 1, 1, 2)$

$A = (0, 1, 0, 2, 1)$

Recursos existentes.

Recursos disponíveis

⑦  $P1; P2; P3; P4$   $R1; R2; R3; R4; R5$

n.º de  
células

C=

0	1	1	1	2	P1
0	1	0	1	0	P2
0	0	0	0	1	P3
2	1	0	0	0	P4

matriz requisição

1	1	0	2	1
0	1	0	2	1
0	2	0	3	1
0	2	1	1	0

N = de recursos e matriz pelo processo

N = de recursos e que o processo i exige

Mostrar se tem impasses. Algoritmo de liberação de impasse

\* P2, P3

- O primeiro a ser atendido será o P2, visto que sua requisição  $(0, 1, 0, 2, 1)$  bate com os recursos disponíveis  $(0, 1, 0, 2, 1)$ , fazendo esta liberação a obtenção  $(0, 1, 0, 1, 0)$  e se acrescenta os recursos disponíveis  $(0, 1, 0, 2, 1) + (0, 1, 0, 1, 0) = (0, 2, 0, 3, 1)$ .

- O segundo a ser atendido é o P3.  $(0, 2, 0, 3, 1) + (0, 0, 0, 0, 1) = (0, 2, 0, 3, 2)$ .

- ~~...~~

P1 P2 P3 P4

A - 0 1 0 2 1

F + T F

A - 0 2 0 3 1

A - 0 2 0 3 2

\* Impasse encontrado.

Impossível prosseguir com os processos P1, P4.

1 → Encontro p. e recursos

2 → Se sim, adiciona allocation a disponíveis.

3 → Se não libera recursos

work + alasan

work  
⑧ Disponivel. → 002 11  
necessidade

PA	0	1	0	0	2
PB	0	2	1	0	0
PC	1	0	3	0	0
PD	0	0	1	1	1

\*PD, PA, PB, PC.

00211 > 01002 ✗  
↓

00211 > 02100 ✗  
↓

00211 > 10300 ✗  
↓

00211 > 00111 ✓<sub>PD</sub>  
↓

11321 > 01002 ✓<sub>PA</sub>  
↓

21532 > 02100 ✓<sub>PB</sub>  
↓

41642 > 10300 ✓<sub>PC</sub>

solução