

## 1. a) Valor "mínimo"

i)

down(1)	down(2)	down(3)
print(A)	print(B)	print(C)
up(3)		up(1)
		up(2)

Sem: 1 2 3  
0 1 1 → Menos que isto não

0	0	1	B
0	0	0	C
0	1	0	A

ii)  
Seria qualquer outra saída possível com a inicialização acima,

## 1. b)

down → Bloqueia se não estiver a 0  
up → é incrementada e guardada

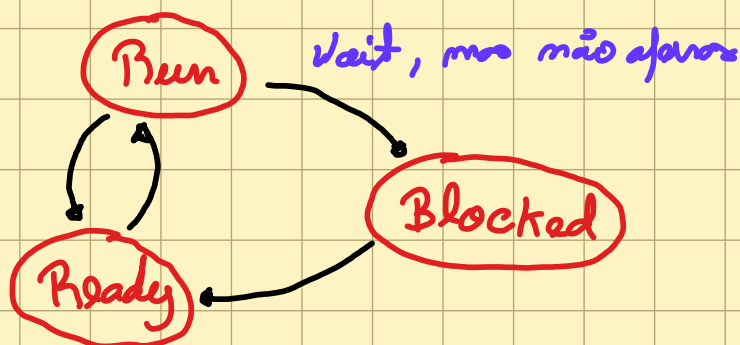
wait

↓  
bloqueia o thread que executou

Signal → Se ninguém ouvir, o valor perde-se

```
lock(me)
while (!cond)
    wait
cond = false
unlock(me)
```

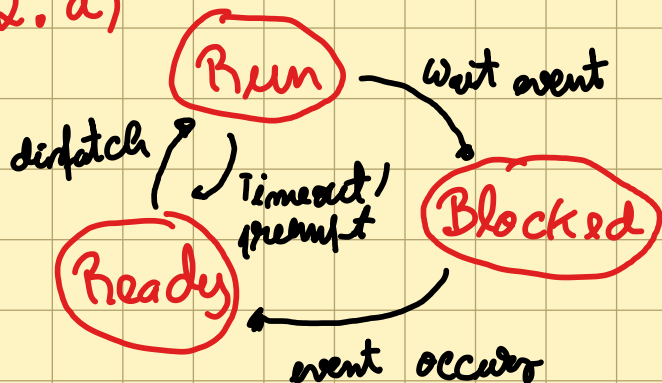
```
lock(me)
cond = true
Signal
unlock(me)
```



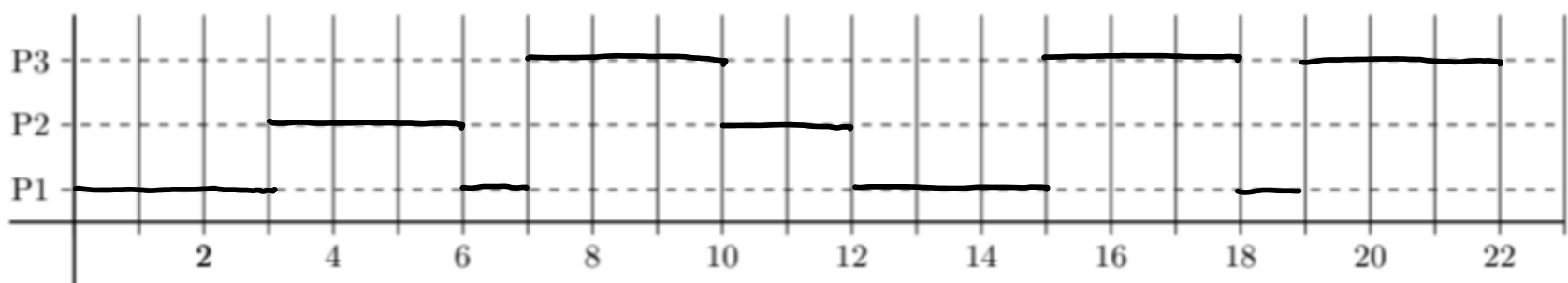
## 1. c)

sem mutex tem race conditions, é por isso que tem um mutex

## 2. a)



2.5)



$t = 0$  :  $P_1 \rightarrow \text{Ready}$

$t = 0$  :  $P_1 \rightarrow \text{Run}$

$t = 2$  :  $P_2 \rightarrow \text{Ready}$

$t = 3$  : (Timeout),  $P_1 \rightarrow \text{Ready}$ ,  $P_2 \rightarrow \text{Run}$

$t = 4$  :  $P_3 \rightarrow \text{Ready}$

$t = 6$  : (Timeout),  $P_2 \rightarrow \text{Ready}$ ,  $P_1 \rightarrow \text{Run}$ ,  $P_3 \rightarrow \text{Ready}$  ( $P_1$  está o mais tempo em Run)

$t = 7$  : (wait  $P_1$ ),  $P_3 \rightarrow \text{Run}$ ,  $P_1 \rightarrow \text{Blocked}$ ,  $P_2 \rightarrow \text{Ready}$

$t = 7.5$  :  $P_1 \rightarrow \text{Ready}$

...

...

$t = 12$  :  $P_2 \rightarrow \text{Termina}$  ...

...

...

3.a) Partições fixas - pré-divisão da memória em blocos

...

3.b)

i) base : start address da memória real onde o pro. foi alojado  
limite : espaço de memória alocado

ii) Dispatch. Escolhe um processo em ready, após consulta a tabela de processos para saber os registos.

3.c)

10000 - OS	
A - 10000	}
B - 40000	
C - (20 000)	
D - 20 000	
10 000	

4.5)

(1, 1, 1)		$P_4 \rightarrow (1, 0, 0) \rightarrow \text{Termine}$
(2, 2, 1)		$P_1 \rightarrow (2, 0, 0) \rightarrow \text{Termine}$
(5, 3, 3)		$P_2 \text{ e } P_3 \rightarrow \text{Podem correr linearmente (estado Safe)}$

Simulo do avali e requis	(1, 1, 1)		$P_3 \rightarrow (3, 1, 0) \text{ mas pede } (0, 1, 0) \rightarrow \text{Posso atribuir?}$
	(1, 0, 1)		
	(2, 1, 1)		
	(5, 2, 3)		
			$P_3 \text{ tem } (1, 2, 0) \text{ e pede pedir } (3, 0, 0)$
			estado requis