

Atividade 7

Deadlock

1. Execute o simulador de deadlock usando o seguinte comando:

java deadlock a 2 2

R = Baixado o deadlock, extraído os arquivos e executado via terminal linux.

Não houve deadlock, pois o estado de deadlock ocorre quando dois ou mais processos estão esperando indefinidamente por um evento que só pode ocorrer por um dos processos em espera, como no caso nenhum dos processos foi bloqueado e sempre os dois estavam disponível ou dois estavam em uso, não ficando nenhum disponível.

```
~$ java deadlock a 2 2
Picked up _JAVA_OPTIONS: -Djava.io.tmpdir=/home/user/tmp -Xms64m
time = 0 available = 2 blocked = 0
time = 1 available = 2 blocked = 0
time = 2 available = 2 blocked = 0
time = 3 available = 2 blocked = 0
time = 4 available = 2 blocked = 0
time = 5 available = 2 blocked = 0
time = 6 available = 2 blocked = 0
time = 7 available = 2 blocked = 0
time = 8 available = 2 blocked = 0
time = 9 available = 2 blocked = 0
time = 10 available = 0 blocked = 0
time = 11 available = 0 blocked = 0
time = 12 available = 0 blocked = 0
time = 13 available = 0 blocked = 0
time = 14 available = 0 blocked = 0
time = 15 available = 0 blocked = 0
time = 16 available = 0 blocked = 0
time = 17 available = 0 blocked = 0
time = 18 available = 0 blocked = 0
time = 19 available = 0 blocked = 0
time = 20 available = 2 blocked = 0
~$ █
```

Figura 1: Execução comando java deadlock a 2 2



Figura 2: Execução do comando no tempo 4

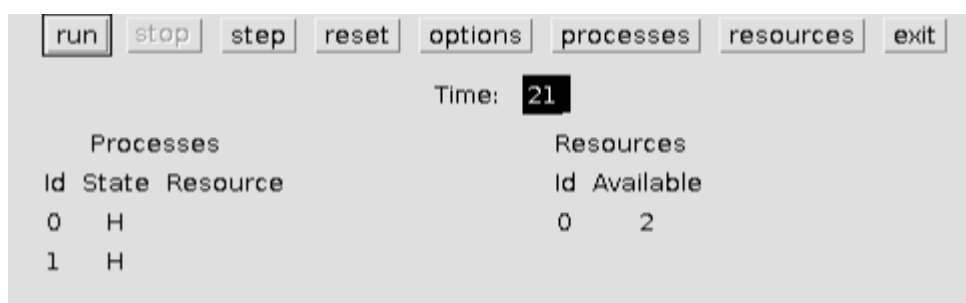


Figura 3: Execução do comando no tempo 21

2. Existem dois arquivos de comando adicionais (b0.dat e b1.dat).

- Execute o simulador de deadlock com o comando:

```
java deadlock b 2 1 1
```

O que aconteceu?

R = Os dois processos só foram bloqueados no último instante da execução e ficaram Disponíveis só em par 1 1 ou 0 0, não ocorre deadlock.

```
~$ java deadlock b 2 1 1
Picked up _JAVA_OPTIONS: -Djava.io.tmpdir=/home/user/tmp -Xms64m
time = 0 available = 1 1 blocked = 0
time = 1 available = 1 1 blocked = 0
time = 2 available = 1 1 blocked = 0
time = 3 available = 1 1 blocked = 0
time = 4 available = 1 1 blocked = 0
time = 5 available = 1 1 blocked = 0
time = 6 available = 1 1 blocked = 0
time = 7 available = 1 1 blocked = 0
time = 8 available = 1 1 blocked = 0
time = 9 available = 1 1 blocked = 0
time = 10 available = 0 0 blocked = 0
time = 11 available = 0 0 blocked = 0
time = 12 available = 0 0 blocked = 0
time = 13 available = 0 0 blocked = 0
time = 14 available = 0 0 blocked = 0
time = 15 available = 0 0 blocked = 0
time = 16 available = 0 0 blocked = 0
time = 17 available = 0 0 blocked = 0
time = 18 available = 0 0 blocked = 0
time = 19 available = 0 0 blocked = 0
time = 20 available = 0 0 blocked = 2
□
```

Figura 4: Execução do comando java deadlock b 2 1 1

- Agora execute o simulador de deadlock com o comando:

```
java deadlock b 2 1 2
```

Executado, como segue o print:

```
~$ java deadlock b 2 1 2
Picked up _JAVA_OPTIONS: -Djava.io.tmpdir=/home/user/tmp -Xms64m

time = 0 available = 1 2 blocked = 0
time = 1 available = 1 2 blocked = 0
time = 2 available = 1 2 blocked = 0
time = 3 available = 1 2 blocked = 0
time = 4 available = 1 2 blocked = 0
time = 5 available = 1 2 blocked = 0
time = 6 available = 1 2 blocked = 0
time = 7 available = 1 2 blocked = 0
time = 8 available = 1 2 blocked = 0
time = 9 available = 1 2 blocked = 0
time = 10 available = 0 1 blocked = 0
time = 11 available = 0 1 blocked = 0
time = 12 available = 0 1 blocked = 0
time = 13 available = 0 1 blocked = 0
time = 14 available = 0 1 blocked = 0
time = 15 available = 0 1 blocked = 0
time = 16 available = 0 1 blocked = 0
time = 17 available = 0 1 blocked = 0
time = 18 available = 0 1 blocked = 0
time = 19 available = 0 1 blocked = 0
time = 20 available = 0 0 blocked = 1
time = 21 available = 0 0 blocked = 1
time = 22 available = 0 0 blocked = 1
time = 23 available = 0 0 blocked = 1
time = 24 available = 0 0 blocked = 1
```

Figura 5: Execução do comando java deadlock b 2 1 2

Por que o resultado do primeiro comando foi diferente do segundo? O que aconteceu de diferente? Explique sua resposta em termos do que está acontecendo nos arquivos de comando: b0.dat e b1.dat.

R = Dessa vez apenas um é bloqueado, no outro caso houve 2. O caso em que houve um bloqueado ocorre apenas quando os disponíveis estão em 0 0, para os casos 1 2 e 0 1 não ocorreu o bloqueio. Ocorre deadlock pois o processo tem dependência do outro.



The screenshot shows a Java deadlock simulation interface. At the top, there are buttons: run, stop, step, reset, options, processes, resources, and exit. The 'stop' button is highlighted. Below the buttons, the 'Time' is displayed as 27. The interface is divided into two main sections: 'Processes' and 'Resources'.

Processes			Resources	
Id	State	Resource	Id	Available
0	C		0	0
1	W	0	1	0

Figura 6: Execução do comando java deadlock b 2 1 2