

git push https://ghp\_Uf2VvVtaY05vQK9mjIYUUnJDLXy44v2gZgZs@github.com:CaioCohen/ComputacaoConcorrente.git

## Lab 5

### Atividade 1:

2-Sim, o programa hellobye.c sempre printou o bye depois dos dois “hello”s.

3-Agora, o programa nem encerrou. Isso aconteceu porque a thread A foi executada antes da thread B começar, além disso, como só tem apenas uma thread A, a variável global x nunca chegou em 2, então o sinal de desbloqueio da thread B nunca foi emitido, então o programa chegou em deadlock.

4 e 5-Agora, algumas vezes o programa entra em deadlock.se as duas threads B forem executadas depois das A, então tudo certo, mas se as Threads B forem executadas antes da variável X chegar em 2, ambas vão ficar bloqueadas, mas como a lógica do algoritmo permite que apenas 1 único sinal de desbloqueio seja mandado, apenas 1 das threads B vai ser desbloqueada.

### Atividade 2:

2-Sim, as threads A só printam Bye depois de B printar HELLO

### Atividade 3:

2-Não, em um dos testes a thread B printou 29.

3-Sim podemos, inclusive, devemos.

4-Sim, dessa vez foi sempre o correto. Isso acontece por quê, antes, com if, ele apenas checava a condição antes de se bloquear, então depois de se desbloquear, é provável que o x se altere e ela não cheque de novo. Com while isso não acontece, ele vai sempre checar.