

WAKEUP





Sleep e Wakeup

Wakeup

- Primitivas de comunicação entre processos.
- Evitam a espera ocupada.
- Abordagem

 Bloquear a execução dos processos quando a eles não é permitido entrar em suas regiões críticas.

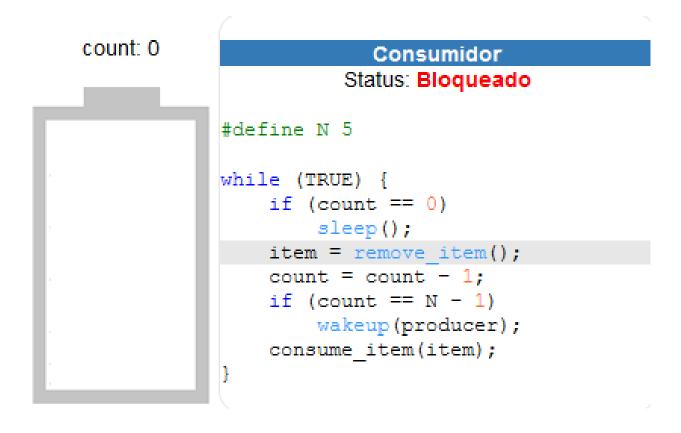
- Sleep Faz com que o processo que a chamou bloqueie, isto é, seja suspenso até que outro processo o desperte.
 - Sinaliza (acorda) o processo anteriormente bloqueado pelo Sleep.
 - Tem como parâmetro o processo a ser desperto.

Produtor-consumidor com Sleep e Wakeup



Consumidor

Dorme quando o buffer está vazio.





Produtor

Dorme quando o buffer está cheio.

```
Produtor
Status: Bloqueado

#define N 5

while (TRUE) {
   item = produce_item();
   if (count == N)
        sleep();
   insert_item(item);
   count = count + 1;
   if (count == 1)
        wakeup(consumer);
}
```

O problema do Sleep e Wakeup

Condição de Corrida: o acesso a count não é restrito.

Sinal Perdido: o produtor tenta acordar o consumidor, mas este não está dormindo.

Podem ocorrer situações em que ambos dormem para sempre.

Exemplo

```
Buffer: [ ]
Count = 0
```

1. Consumidor verifica se o buffer está vazio.

```
Buffer: [ ]
Count = 0
```

Produtor

- 2. insere um item no buffer-
- 3. incrementa count para 1.-

```
Buffer: [ X ]
Count = 1
```

Consumidor:

```
Enquanto verdadeiro:
```

```
se count == 0:
    dormir()
item = buffer.remover()
count = count - 1;
se (count == N - 1)
    acordar_produtor()
consumir_item(item);
```

Produtor:

```
Enquanto verdadeiro:
    item = produzir_item()
    se count == N:
        dormir()

buffer.adicionar(item)
    count = count + 1;
    se count == 1:
        acordar_consumidor()
```

Exemplo

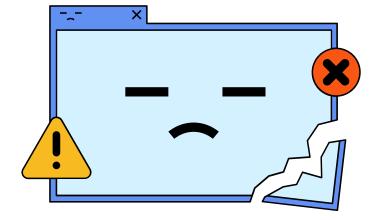
4. Produtor verifica se count = 1 e tenta acordar o **consumidor**.

```
Buffer: [ X ]
Count = 1
```

- **5.** O **consumidor** não está dormindo e o sinal é perdido.
- **6.** Quando o **consumidor** volta a ser executado, executa *sleep* e é bloqueado.
- 7. Produtor ganha a CPU e, ao encher o buffer, irá dormir também.

```
Consumidor:
    Enquanto verdadeiro:
        se count == 0:
            dormir()
        item = buffer.remover()
        count = count - 1;
        se (count == N - 1)
             acordar_produtor()
        consumir_item(item);
Produtor:
    Enquanto verdadeiro:
        item = produzir_item()
        se count == N:
            dormir()
        buffer.adicionar(item)
        count = count + 1;
        se count == 1:
             acordar consumidor()
```

Simulação Sleep e Wakeup



```
count: 0
               Produtor
                                                                                           Consumidor
            Status: Pronto
                                                                                          Status: Pronto
#define N 5
                                                                              #define N 5
                                                                             while (TRUE) {
while (TRUE) {
    item = produce item();
                                                                                  if (count == 0)
    if (count == N)
                                                                                      sleep();
        sleep();
                                                                                  item = remove item();
    insert item (item);
                                                                                  count = count - 1;
    count = count + 1;
                                                                                  if (count == N - 1)
    if (count == 1)
                                                                                      wakeup (producer);
                                                                                  consume_item(item);
        wakeup (consumer);
                                                                                        Executar Consumidor
           Executar Produtor
                                                      Desfazer
```

http://lasdpc.icmc.usp.br/~ssc640/grad/bcc2015/grupoa12/simulation.html



Simulação Sleep e Wakeup

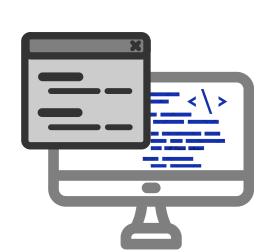
■ Simulador - Produtor-Consumidor com Sleep e Wakeup



1 # Define N 4		Consumidor 1 # Define N 4
2 3 Enquanto verdadeiro: 4 item = produzir_item() 5 se count == N: 6 dormir() 7 buffer.adicionar(item) 8 count = count + 1 9 se count == 1: 10 acordar_consumidor()	Buffer Count: 0	2 3 Enquanto verdadeiro: 4 se count == 0: 5 dormir() 6 item = buffer.remover() 7 count = count - 1 8 se count == N - 1: 9 acordar_produtor() 10 consumir_item(item)
Pronto!		Executar Consumidor Pronto!

Simulador: Produtor-consumidor com Sleep e Wakeup

https://github.com/software-concorrente-distribuido/art-attack/tree/main/SleepWakeupSimulator



Referência

TANENBAUM, A.; BOS, H. Sistemas Operacionais Modernos. 4ª Edição. São Paulo: Pearson Education do Brasil, 2016.