


© 2004-2010 Volnys Bernal 1

Exercício programa: Produtor Consumidor com Mutex e Fila

Volnys Borges Bernal
volnys@lsi.usp.br
<http://www.lsi.usp.br/~volnys>



© 2004-2010 Volnys Bernal 2

EP: Produtor consumidor com mutex e fila

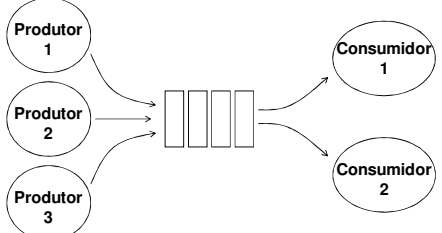
□ **Objetivo do trabalho:**

- ❖ Implementar a resolução do problema produtor-consumidor utilizando:
 - As primitivas de exclusão mútua da biblioteca pthreads.
 - A implementação de fila com uma quantidade limitada de slots

© 2004-2010 Volnys Bernal 3

EP: Produtor consumidor com mutex e fila

□ **Problema do produtor-consumidor:**



© 2004-2010 Volnys Bernal 4

EP: Produtor consumidor com mutex e fila

```

Produtor ()
{
  repetir
  {
    Produzir (E);

    lock ();
    enquanto FilaCheia (F)
    {
      unlock ();
      lock ();
    }
    InserirFila (F,E);
    unlock ();
  }
}

Consumidor ()
{
  repetir
  {
    lock ();
    enquanto FilaVazia (F)
    {
      unlock ();
      lock ();
    }
    E = RetirarFila (F);
    unlock ();

    Processar (E);
  }
}

```

© 2004-2010 Volnys Bernal 5

EP: Produtor consumidor com mutex e fila

□ **Utilizar as seguintes primitivas de exclusão mútua do pthreads :**

```

int  pthread_mutex_init  (pthread_mutex_t *mutex, NULL);
int  pthread_mutex_init  (pthread_mutex_t *mutex,
                          pthread_mutexattr_t *attr)
int  pthread_mutex_lock  (pthread_mutex_t *mutex)
int  pthread_mutex_unlock (pthread_mutex_t *mutex)
int  pthread_mutex_trylock (pthread_mutex_t *mutex)

```

© 2004-2010 Volnys Bernal 6

EP: Produtor consumidor com mutex e fila

□ **Utilizar as seguintes primitivas de fila:**

```

int  InitFila (struct fila *F)
int  FilaVazia (struct fila *F)
int  FilaCheia (struct fila *F)
void InserirFila (struct fila *F, int item)
int  RetirarFila (struct fila *F)

```

© 2004-2010 Volnys Bernal 7

EP: Produtor consumidor com mutex e fila

□ **Implementação de Fila**

<p>RetirarFila(F) Se FilaVazia(F) retornar(ERRO)</p> <p>Senão item = F.buffer[F.inicio]; F.inicio = (F.inicio + 1) mod F.tam F.Nitens = F.nitens - 1; retornar(item);</p> <p>InserirFila(F,elemento) Se FilaCheia(F) retornar(ERRO)</p> <p>Senão F.buffer[F.fim] = elemento; F.fim = (F.fim + 1) mod F.Tam; F.nitens = F.nitens + 1</p>	<p>InitFila(F) Inicio=0; Fim = 0; Itens = 0;</p> <p>FilaVazia(F) Se F.itens == 0 Retornar(V)</p> <p>Senão Retornar(F)</p> <p>FilaCheia(F) Se F.itens == F.tam Retornar(V)</p> <p>Senão Retornar(F)</p>	<table border="1" style="border-collapse: collapse; text-align: center;"> <tr> <td style="padding: 2px;">Buffer</td> <td style="padding: 2px;">0</td> <td style="padding: 2px;">1</td> <td style="padding: 2px;">2</td> <td style="padding: 2px;">3</td> <td style="padding: 2px;">4</td> <td style="padding: 2px;">5</td> </tr> <tr> <td style="padding: 2px;"></td> <td style="padding: 2px;"></td> <td style="padding: 2px;">X</td> <td style="padding: 2px;">X</td> <td style="padding: 2px;"></td> <td style="padding: 2px;"></td> <td style="padding: 2px;"></td> </tr> <tr> <td style="padding: 2px;"></td> <td style="padding: 2px;">2</td> <td style="padding: 2px;">4</td> <td style="padding: 2px;">2</td> <td style="padding: 2px;">6</td> <td style="padding: 2px;"></td> <td style="padding: 2px;"></td> </tr> <tr> <td style="padding: 2px;"></td> <td style="padding: 2px;">Inicio</td> <td style="padding: 2px;">Fim</td> <td style="padding: 2px;">Nitens</td> <td style="padding: 2px;">Tam</td> <td style="padding: 2px;"></td> <td style="padding: 2px;"></td> </tr> </table>	Buffer	0	1	2	3	4	5			X	X					2	4	2	6				Inicio	Fim	Nitens	Tam		
Buffer	0	1	2	3	4	5																								
		X	X																											
	2	4	2	6																										
	Inicio	Fim	Nitens	Tam																										

© 2004-2010 Volnys Bernal 8

EP: Produtor consumidor com mutex e fila

□ **Para compilar**

```
cc -o prodcons prodcons.c fila.c -lpthread
```

Arquivo de saída executável: prodcons
 Arquivos fonte: prodcons.c fila.c

© 2004-2010 Volnys Bernal 9

EP: Produtor consumidor com mutex e fila

□ **Trabalho:**

- ❖ Grupo de 2 pessoas
- ❖ Data entrega: 10/04
- ❖ O trabalho deve conter:
 - Página de rosto informando:
 - Nome da disciplina
 - Título do trabalho
 - Nome dos autores
 - Objetivo
 - Descrição
 - Listagem do programa
- ❖ **Formato do trabalho impresso:**
 - Papel A4
 - Folhas grampeadas (não encadernar!)
- ❖ **Entrega:**
 - Entrega na sala de aula do trabalho impresso
 - Execução do programa em sala de aula
 - Serão descontados 2 pontos da nota para cada dia de aula (semana) em atraso