

Programação Pthreads (variáveis de condição)

Dr. Osmar Marchi dos Santos

Variáveis de Condição

- Permite que uma thread espere, dentro de uma região crítica, até que uma condição sobre algum dado seja modificada
- Efetivamente, permite que as threads deixem de realizar “*busy waiting*” sobre um dado, aumentando a utilização do processador
- Por exemplo (considerando os filósofos glutões):
 - `while (garfo[i] != 0) { }; // busy waiting`
 - `while (grafo[i] != 0) { espere(); }; // var. condição`

Variáveis de Condição

- Uma variável de condição sempre estará relacionada a um mutex
- Ou seja, a condição para esperar na variável de condição, necessita que a thread já tenha obtido o mutex
- Por exemplo:
 - `mutex_lock();`
 - `while (garfo[i] != 0) espere();`
 - `mutex_unlock();`

Variáveis de Condição

- Para usar variáveis de condição temos:

```
#include <pthread.h>
```

```
int pthread_cond_wait(pthread_cond_t *cond, pthread_mutex_t *mutex);
```

```
int pthread_cond_timedwait(pthread_cond_t *cond, pthread_mutex_t *mutex, const struct  
timespec *abstime);
```

```
int pthread_cond_signal(pthread_cond_t *cond);
```

```
int pthread_cond_broadcast(pthread_cond_t *cond);
```

```
int pthread_cond_destroy(pthread_cond_t *cond);
```

```
int pthread_cond_init(pthread_cond_t *restrict cond, const pthread_condattr_t *restrict attr);
```

```
pthread_cond_t cond = PTHREAD_COND_INITIALIZER;
```

Exercícios - Modificar a solução dos filósofos glutões para utilizar variáveis de condição

- Cinco filósofos estão sentados em uma mesa redonda
- Cada filósofo possui um prato à sua frente
- Cada filósofo possui um garfo à sua direita e à sua esquerda
- Um filósofo precisa de dois garfos para conseguir comer
- Cada garfo é um recurso compartilhado
- Os filósofos tem um comportamento cíclico:
 - Pensa
 - Pega garfo à Esquerda
 - Pega garfo à Direita
 - Come
 - Libera garfo à Direita
 - Libera garfo à Esquerda

