

Exemplo de Relatório de SOP

Rafael R. Obelheiro¹

¹DCC – UDESC

rafael.obelheiro@udesc.br

1. Introdução

Este documento ilustra o conteúdo esperado do relatório do trabalho de SOP. A especificação do trabalho estipula que

O relatório deve descrever de que forma foi implementado o controle de concorrência na aplicação, e qual a API utilizada.

A próxima seção descreve o controle de concorrência e a API utilizada no programa `condvar.c`, que é um dos exemplos de IPC com Pthreads mostrados em sala de aula. Nesse programa a *thread* principal (`main()`) cria três *threads*: duas dessas *threads* incrementam uma variável compartilhada `count` (função `inc_count()`), enquanto a terceira *thread* monitora o valor de `count` (função `watch_count()`). Quando `count` atinge um limite predefinido, uma das *threads* que incrementam sinaliza a *thread* que espera, que acorda e modifica `count`. O programa continua até que as duas primeiras *threads* façam `TCOUNT` incrementos. O programa principal imprime o valor final de `count`.

2. Controle de Concorrência

O programa utiliza a API de Pthreads, mais especificamente mutexes e variáveis de condição.

Nesse programa, o estado compartilhado é representado pela variável `count`. Para garantir a exclusão mútua no acesso a `count`, é usado um mutex, implementado pela variável `count_mutex`.

A condição de sincronização é que, depois de serem realizados `COUNT_LIMIT` incrementos, a *thread* que executa `watch_count()` deve ser sinalizada por uma das *threads* que executam `inc_count()`. Essa sincronização é implementada pela variável de condição `count_threshold_cv`.

3. Observações

O seu relatório deve descrever todas as condições de exclusão mútua e sincronização da aplicação, e como elas foram tratadas. Inclua ainda uma explicação de como foi implementada a lista de lanches disponíveis e a comunicação entre as *threads* atendentes e a *thread* de caixa.