



Universidade da Beira Interior : Sistemas Operativos
2º Trabalho - 2º Semestre – 2018/2019

MultiThreaded e IPC Programação

Versão 1.1 17/5/2019

OBJETIVOS

O objetivo deste trabalho é o desenvolvimento em primeiro lugar duma estrutura de dados sincronizado para depois ser usado num servidor aplicacional e os seus clientes. A comunicação entre os clientes e o servidor multithreaded deverá utilizar um mecanismo de IPC (Inter-Process Communication), Pipes com Nomes.

Linguagens de Programação : C, C++

DESCRIÇÃO

A: Implementação e Especificação duma Estrutura de Dados

Primeira Parte

Desenvolvimento duma estrutura de dados clássico, uma lista ligada (single threaded-sem necessidade de sincronização)

Podem usar uma fila simples, circular, duplamente ligada.

Deverá especificar a Estrutura (Abstrata) de Dados e criar Programas de Teste.

Os nós da fila podem ser abstratos (void *) mas pode ser útil considerados nós da fila com o seguinte tipo

- msgId : Integer
- content : String

Segunda Parte

Faça uma ou mais implementações Multithreaded

Deverá especificar o Interface e as situações de concorrência e sincronização resolvidos / não resolvidos com a sua implementação estrutura de dados

Deverá incluir Programas de Teste.

O seu relatório devia incluir uma bibliografia completa das referencias e sites consultados e um resumo das funcionalidades da sua implementação.

B: Desenvolvimento do Servidor Aplicacional

O servidor tem dois objetivos: (i) Um servidor de ficheiros e (2) Uma aplicação onde clientes podem inserir, visualizar e apagar mensagens contidas em filas com a estrutura da parte A.

1 - Clientes Podem Pedir Ficheiros

- Apenas ficheiros em baixo da raiz do servidor (evitar ../../secretFile)
- Ficheiros num subpasta (secret) apenas podem ser acedidos conforme um password.
 - Implemente um protocolo da sua escolha

2 - Clientes Devem Poder Fazer :

- Saber as filas existentes (as filaID's)
- Criar uma nova fila.
- Saber as Mensagens duma determinada fila (as MsgId's dum fila)
- visualizar, inserir e apagar as mensagens
 - Implemente um protocolo da sua escolha

3 – Comunicação entre os clientes e o servidor deverá utilizar IPC (NAMED PIPES)

Nota: O servidor terá de usar multithreading para atender os clientes das filas em simultânea.

ASPECTOS IMPORTANTES

A inscrição do grupo é feita num Formulário do Google,

Cada grupo do trabalho pode ter até 3 elementos do mesmo turno pratico

A inscrição deverá ser feita com o formulário de google (url em baixo) até sexta feira dia 10 de maio.

<https://forms.gle/qZpsRaHbjtaC6Nkw5>

Entregar:

No Moodle: (1) Código (scripts e ficheiros necessários) (2) Relatório em formato digital (pdf)

(3) Relatório Impresso (durante a apresentação)

O relatório (a primeira página deverá indicar os nomes dos elementos do grupo) deverá incluir um breve resumo e explicação do funcionamento do programa (formatos dos ficheiros usados, screenshots dos menus etc.) e indicar os principais pontos da discussão.

Máximo de 15 folhas A4 (30 páginas frente e verso)

INFORMAÇÃO

A data limite de entrega é 23H do dia 7 de junho de 2019

- Entrega até dia 3 de junho 09H: Apresentação do trabalho é a semana de 3-8 de junho
- Entrega até dia 7 de junho 23H : Apresentação do trabalho dias 11 e 12 de junho

Bibliografia

- PThreads Programming : A POSIX Standard for Better Multiprocessing By Dick Buttlar, Jacqueline Farrell, Bradford Nichols, O'Reilly Media <http://www.di.ubi.pt/~operativos/resource/PthreadsProgram/toc.html>
- Sebenta das Aulas Praticas - <http://www.di.ubi.pt/~operativos/praticos/pdf/11-ipc.pdf>
- M. Michael and M. Scott. "Nonblocking algorithms and preemption-safe locking on multiprogrammed shared - memory multiprocessors." Journal of Parallel and Distributed Computing, 1998.
<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.93.1892&rep=rep1&type=pdf>
- C++ Concurrency in Action. A Williams
https://www.bogotobogo.com/cplusplus/files/CplusplusConcurrencyInAction_PracticalMultithreading.pdf

Exemplo

No Server

Fila com (id=1) → 5, "amanha" → 3, "hoje"

Fila com (id=2) → 3, "so" → 3, "5"

Num ou Mais Clientes Concorrentes

Cliente X pede O conteúdo das mensagens nas filas 1 e com id=3

Cliente Y apaga mensagens com id=3 na fila com id=2

etc