



Universidade do Minho
Escola de Engenharia

MESTRADO INTEGRADO EM ENGENHARIA DE
TELECOMUNICAÇÕES E INFORMÁTICA

SISTEMAS OPERATIVOS

AURRAS: PROCESSAMENTO DE FICHEIROS DE ÁUDIO

RELATÓRIO DE PROJETO
GRUPO 1

Trabalho realizado por:

Guilherme Sobrinho - 84674

Pedro Pando - A82431

Renato Gonçalves - A85134

Índice

1	Introdução	1
2	Cliente	2
3	Servidor	2
3.1	Pedidos	2
3.1.1	Transform	2
3.1.2	Funcionalidades do Transform:	2
3.1.3	Status	3
4	Funções Principais	4
4.1	Execute Transform	4
4.2	Execute Status	4
5	Conclusão	5

1 Introdução

No âmbito da unidade curricular de Sistemas Operativos do Mestrado Integrado em Engenharia de Telecomunicações e Informática foi proposto aos discentes desenvolver um serviço de transformação de ficheiros áudio pela aplicação de filtros, este serviço deve ser constituído por um servidor e um por cliente que deverão estabelecer comunicação entre si.

Este serviço terá também a funcionalidade de mostrar todos os pedidos que está a utilizar e a disponibilidade deste.

2 Cliente

Na parte do Cliente, em primeiro lugar temos a comunicação estabelecida com o servidor através de pipes com nome "fifo_leitura" e "fifo_escrita".

O Cliente envia por argumento para o servidor qual tipo de pedido quer solicitar: "transform" e "status". Sendo que, o "status" irá enviar para a interface do Cliente (terminal) o resultado do pedido solicitado.

3 Servidor

O Servidor cria os pipes com nome, um para leitura e outro para escrita, "fifo_leitura" e "fifo_escrita" respetivamente.

Este irá tratar os pedidos solicitados pelo Cliente havendo a criação de processos filhos para cada pedido, e dependendo do tipo de pedidos a nova criação de processos filhos (segunda geração de filhos).

3.1 Pedidos

3.1.1 Transform

Neste pedido o Cliente solicitará ao servidor, a funcionalidade principal do nosso projeto, sendo esta a filtragem escolhida para o ficheiro de áudio introduzido por ele. Este pedido irá criar um ficheiro (xxxxx.mp3) cujo seu nome é introduzido pelo Cliente ao solicitar o pedido, este ficheiro está a ser criado e guardado na pasta "src".

3.1.2 Funcionalidades do Transform:

- Caso o número de filtros solicitado for maior que o número de filtros disponíveis, o pedido não será processado. O programa só executará a filtragem se o número de filtros estipulado de cada pedido for igual ou menor ao do ficheiro de configuração.
- Caso o número de filtros for menor ou igual ao do ficheiro de configuração e menor ou igual aos disponíveis, o programa executará e obteremos o output desejado.
- Caso o número de filtros for menor ou igual ao do ficheiro de configuração e maior que os disponíveis não deixa processar o pedido solicitado pelo Cliente. Não tendo implementado uma fila de espera de pedidos. Quando temos um Cliente a solicitar filtros em utilização o

programa não permite a realização desse pedido tendo que esperar que o pedido anterior termine, não voltando a executar, o Cliente que fez o segundo pedido.

3.1.3 Status

Neste pedido o Cliente solicitará ao servidor, a funcionalidade secundária que visualiza todos os pedidos que o servidor está a tratar e a utilização e disponibilidade dos filtros.

4 Funções Principais

4.1 Execute Transform

Esta função recebe: o ficheiro de escrita (output), o pedido completo, um contador (número de filtros) e o ficheiro de configuração. Compara os filtros que recebe no pedido, cria um filho para cada filtro, de seguida cria um pipe anónimo para cada filtro comunicar com o filtro seguinte. No fundo a função vê quais são os filtros que vêm no pedido enviado, executa os filtros um a um encadeados por pipes anónimos, sendo que quando estes filhos morrem, o pai morre e envia um sinal com pid dele para o avô. O pedido espera que todos os filtros terminem a sua execução para o próprio terminar. A utilização do pid que é enviado para o avô, serve para incrementar os filtros por ele utilizados, isto é, antes de executarmos a função guardamos numa struct (struct "Process") o pid do pedido e toda a informação desse pedido para que quando um processo morra haja atualização dos filtros utilizados. O servidor só entra nesta função quando o número de filtros é menor ou igual aos disponíveis. O pedido é tudo o que o cliente fornece ao servidor.

4.2 Execute Status

Esta função envia para a interface do cliente (terminal) os pedidos dos diferentes clientes e a disponibilidade dos filtros. Nesta função voltamos a aceder à struct "Process" para acedermos a todos os pedidos solicitados pelos diferentes clientes, e para podermos ver a utilização dos filtros pelos clientes. A disponibilidade dos filtros só é visível caso o pedido possa ser processado, isto é, se o pedido solicitar o número de filtros maior que o disponível, esse pedido não é mostrado por esta função.

5 Conclusão

Ao longo da realização deste projeto deparámo-nos com algumas dificuldades, tanto a nível de conhecimento da matéria para o desenvolvimento do projeto, como da melhor maneira de implementação da solução para este problema.

Depois de vários dias de trabalho e com o esclarecimento de dúvidas elucidadas pelos docentes, conseguimos implementar a solução, sendo uma mais valia também como estudo para a avaliação teórico-prática.

Podemos definir este projeto prático como um sucesso visto que os objetivos que nos foram propostos foram atingidos na sua plenitude.