Relatório Sistemas Operativos

Instituto Superior de Engenharia de Coimbra

Engenharia Informática

João Bernardo Teixeira Baptista – 2020131684

Pedro Miguel Dinis Sequeira – 2020132079

Índice

Header file medicalso.h

Contêm as estruturas de dados cliente e medico, assim como algumas funções que são utilizadas várias vezes ao longo do programa, encerrar, encerrar_cliente, trataSinal_cliente, trataSinal_medico e trataAlarm que dão unlink a FIFOS que comunicam com o balcão.

Dentro da estrutura cliente temos o nome do mesmo (char nome[10]); os seus sintomas (char sintomas[100]); a especialidade em que estes se encontram (char especialidade[15]; a prioridade desses sintomas (int prioridade); tipo, que indica de onde vem uma mensagem, no caso do cliente o tipo é 'c'; estado, uma flag que se encontra a 0 se não estiver em consulta e a 1 se estiver; e o seu pid)

Dentro da estrutura medico encontra-se o nome do mesmo (char nome[10]); a sua área ou especialidade (char especialidade[15]); tipo, que indica de onde vem uma mensagem, no caso do médico o tipo é 'm'; estado, uma flag que se encontra a 0 se não estiver em consulta e a 1 se estiver; e o seu pid.

Balcao.c

No ficheiro c relativo ao balcão são no início da função main atribuídos os valores para as variáveis de ambiente MAXCLIENTES E MAXMEDICOS através do processo getenv, que estabelecem o número máximo de clientes e médicos, respetivamente, que podemos ter em simultâneo no programa.

Em seguida são criados o fifo ligado ao balcão (BALCAO_FIFO), as variáveis b2c (balcão to classificador) e c2b (classificador to balcão) e os seus respetivos pipes, são também declarados char mensagem[100],que contém os sintomas introduzidos pelo cliente, resposta[100], que armazena a mensagem vinda do médico, armazena a .

É então feito um fork que cria dois processos (pai e filho); no processo pai é fechado o lado de leitura do pipe b2c (close(b2c[0]))e o de escrita do pipe c2b (close(c2b[1])).

No processo filho são fechadas as outras extremidades dos pipes, respetivamente b2c e c2b (close(b2c[1]), close(c2b[0])) e duplicadas as extremidades abertas (dup(b2c[0]), dup(c2b[1])) e fechar as antigas (close(b2c[0]), close(c2b[1])).

É então executado o programa classificador (execl("classificador", "classificador", NULL)), sendo que se este não funcionar por algum motivo o programa balcão é encerrado(exit(-1)).

No processo pai é feito um select que deteta mensagens da consola, podendo assim identificar vários comandos, como "encerrar" que encerra o balcão, encerrando assim simultaneamente os programas medico ou cliente; "utentes", que lista todos os utentes, quer

estejam em lista de espera ou em consulta; “especialistas”, que lista todos os médicos, em consulta ou não; “delut X”, que remove o utente X do sistema e “delesp X”, que remove o médico X do sistema, qualquer outro comando é considerado inválido.

É comunicada ao balcão a chegada de um novo utente, assim como a sua saída, o mesmo se aplica para o médico.

Cliente.c

O programa cliente é inicializado com o nome do mesmo, esse nome é registado, assim como o pid de cada cliente e em seguida são perguntados os seus sintomas.
Após receber os sintomas o utente aguarda pela chegada de um médico, podendo sair do programa ao escrever “d”, quando eventualmente um aparecer é estabelecida uma ligação entre ambos através dos fifos MEDICO_FIFO_FINAL e CLIENTE_FIFO_FINAL, assim o médico e utente conseguem comunicar através de um sistema select.

Se o balcão encerrar o cliente é igualmente encerrado.

Medico.c

O programa medico é inicializado com o nome no mesmo e a sua especialidade, estes dados ficam guardados, assim como o pid do médico.
O médico espera que lhe seja atribuído um cliente e pode sair do programa escrevendo “sair”.

Através de um select o médico consegue comunicar com um utente e vice-versa.

Se o balcão encerrar o médico é igualmente encerrado.

Problemas Encontrados

Proteção de read/write

Durante a realização do trabalho foi comum a utilização de reads e writes, no entanto estes não foram protegidos com o código habitual de “if(read/write(…)==-1){exit(-1);}”.

Acontece que após o trabalho se encontrar no seu estado final a inclusão destas linhas de código fez com que não funcionasse, não sabemos o porquê de isto acontecer então foi tomada a decisão de excluir esse código e voltar a um anterior, funcional.

Sistema “Sinal de Vida”

No enunciado é descrito um sistema que avise o balcão da existência de um médico a cada 20 segundos; ao tentar implementar esta funcionalidade com um alarm(20) não houve sucesso, então esta parte do código encontra se comentada.