**SISTEMAS OPERATIVOS**

Trabalho prático

Meta 2 - (2018/2019)

**Turma:** P8

**Trabalho realizado por**:

João Correia : 21250416

Luís Freitas : 21270475

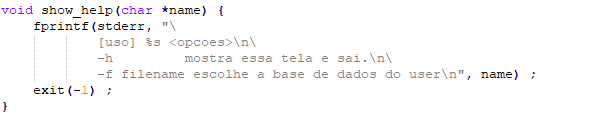
**Índice**

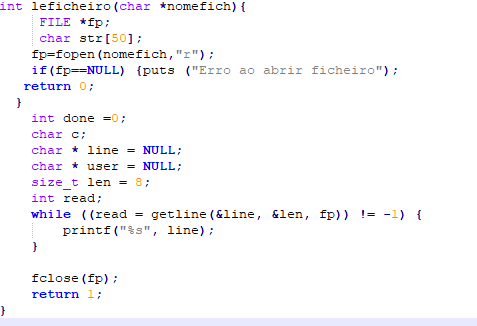
Servidor 3

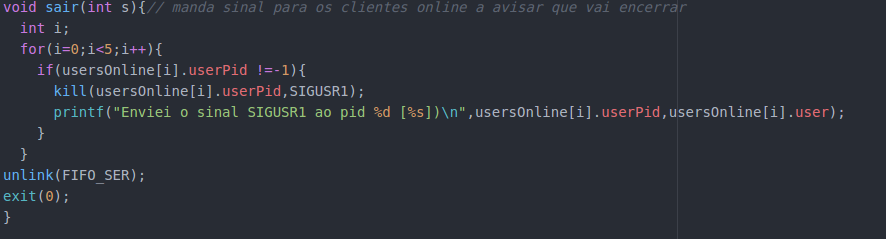
Cliente............................................................................................................................................12

**Servidor:**

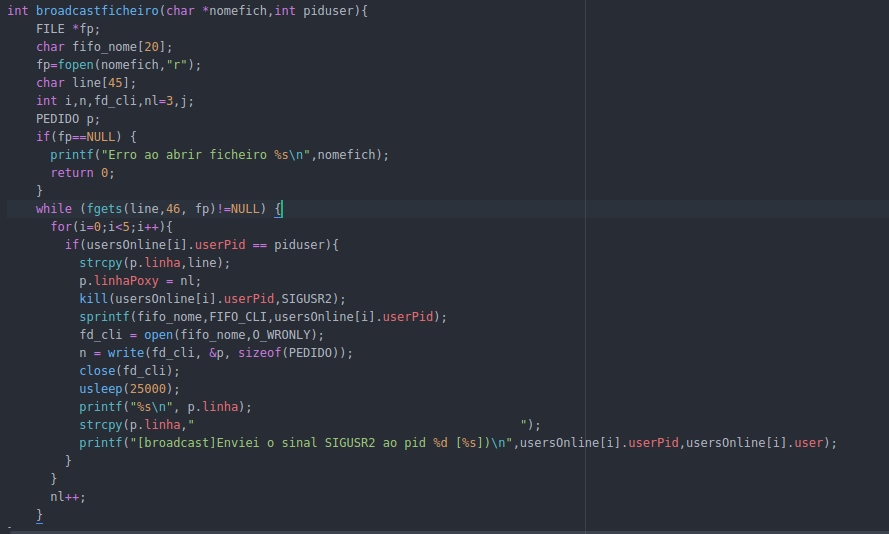
A função void show\_help(char \*name) mostra as opções que existem, o name serve para mostrar o path de uma maneira simples. Quando iniciado com -h irá mostrar a tela de opções.



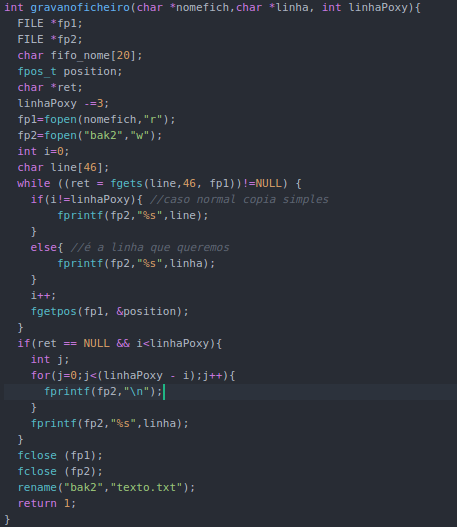
Int leficheiro(char \*nomefich) serve para ler o ficheiro e verificar se é lido ou não, returnando 0 se não e 1 se sim.  


Void sair(int s) ao comando **shutdown** manda para todos os users ligados um sinal a avisar que o servidor encerrou e encerra os clientes.

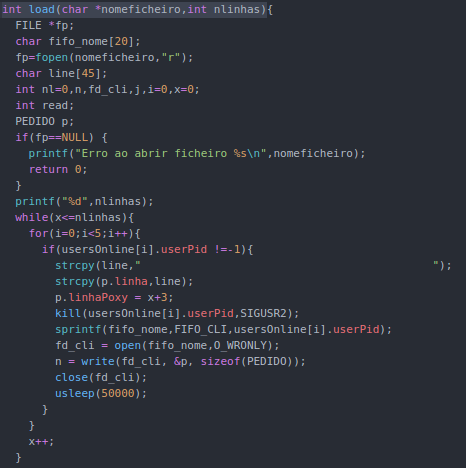
int **broadcastficheiro**(char \*nomefich,int piduser) atualiza a nova linha editada por um user para todos os outros users.

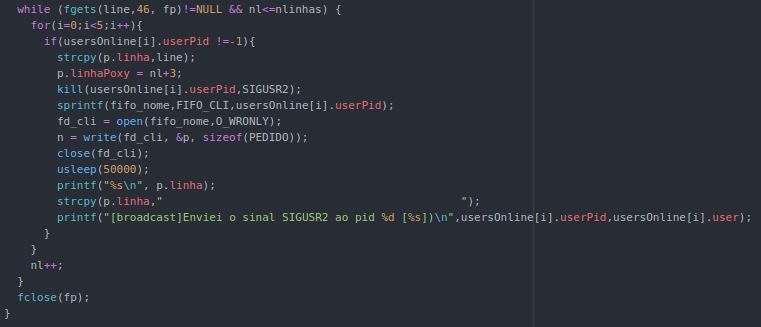


int **gravanoficheiro**(char \*nomefich,char \*linha, int linhaPoxy), guarda num ficheiro a nova linha, sempre que existe um unlockline(ou seja, quando um user passa do modo de edição e passa para o modo de navegação).



int **load**(char \*nomeficheiro,int nlinhas), ao comando load vai limpar o ecrã, pedir um ficheiro e carrega para os users o novo ficheiro de edição.

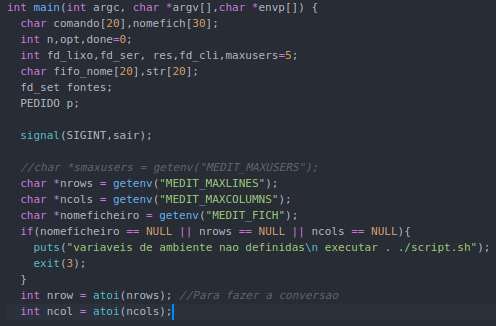




int **checauser**(char \*nomefich,char \*username), recebe o user que dá login no cliente, e verifica se está no ficheiro de texto dos usernames, se sim dá return de 1 ou 0 se não.



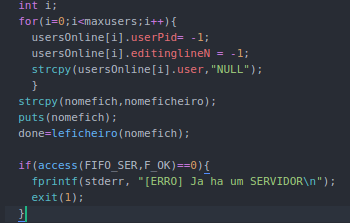
Int main(),neste pedaço de código verificamos se o script das variáveis ambientes foi corrido e avisa, e se não tiver sido corrido previamente encerra.



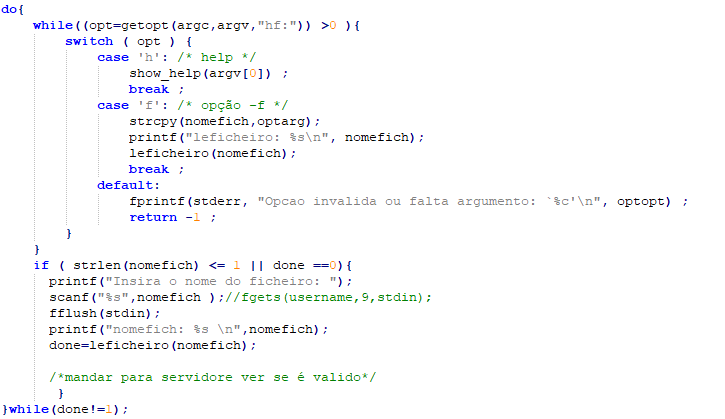
Criamos um fork para o filho que vai estar a correr o aspell.



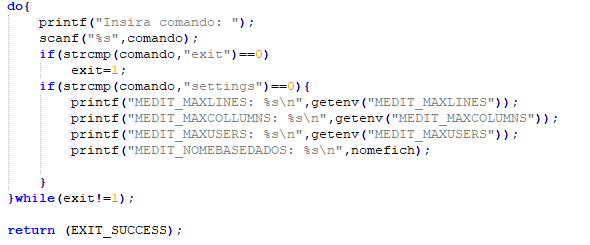
Nesta parte incializamos uma tabela de users possíveis de serem registados, e verificamos se o servidor já está a ser corrido.



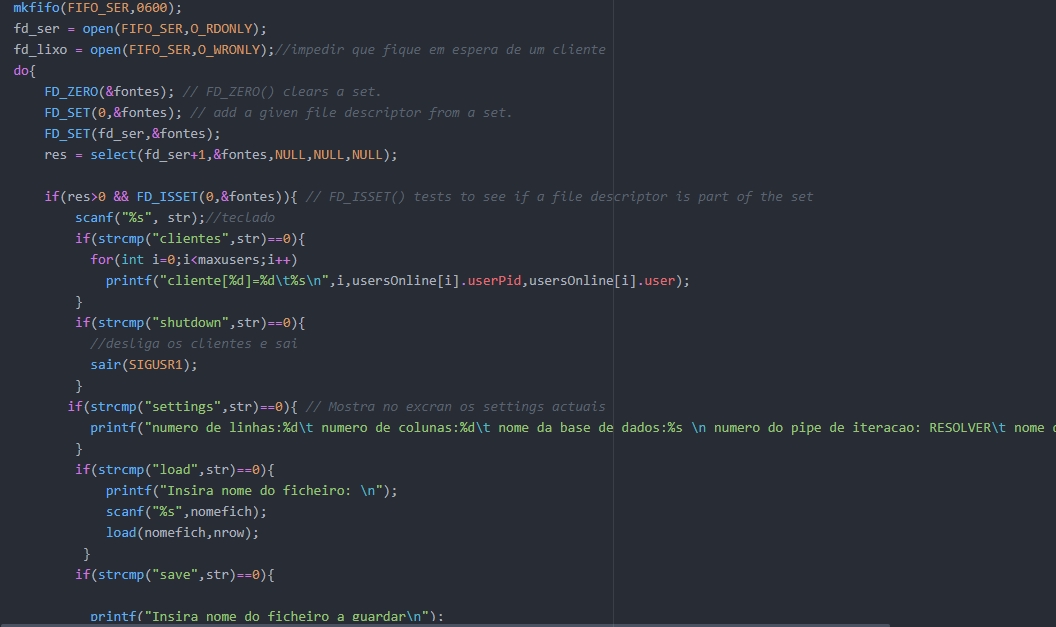
Recebe os comandos h e f, e vai continuar a corre r enquanto não receber um ficheiro válido.



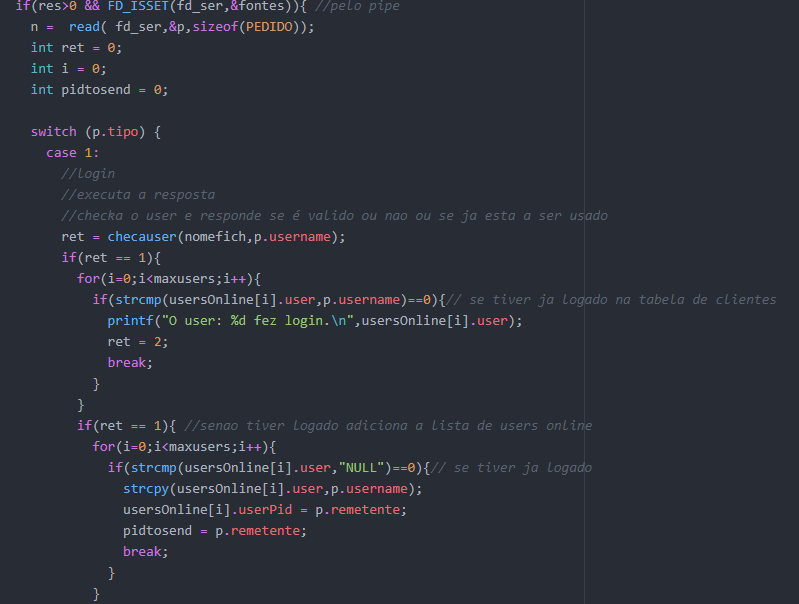
Continua a executar enquanto não recebe comandos válidos, e mostra a opção settings ou a saída do server.



Criamos o fifo do servidor, fazemos os selects e verificamos as settings escolhidas pelo servidor.

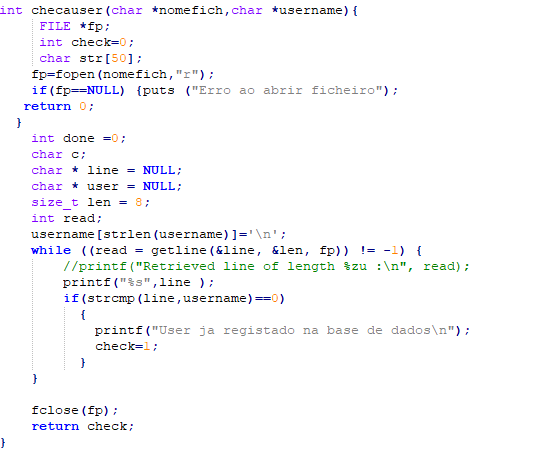


Recebe pelo select do pipe, valida se o user está no ficheiro de texto, se estiver adiciona à tabela de users que podem usar a edição.

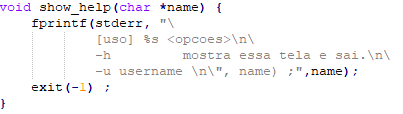


**Cliente:**

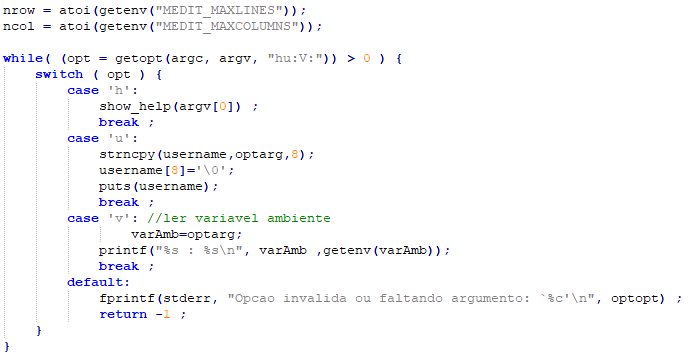
Int checauser (char \*nomefich,char \*username), abre o ficheiro de base de dados, e verifica se o user inserido na main se encontra na mesma.



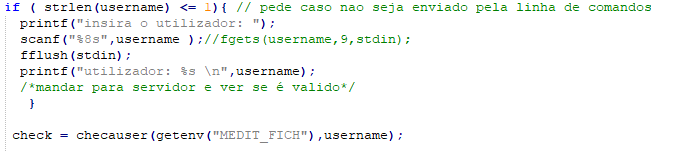
Mostra as opções possíveis de comandos a serem introduzidos.



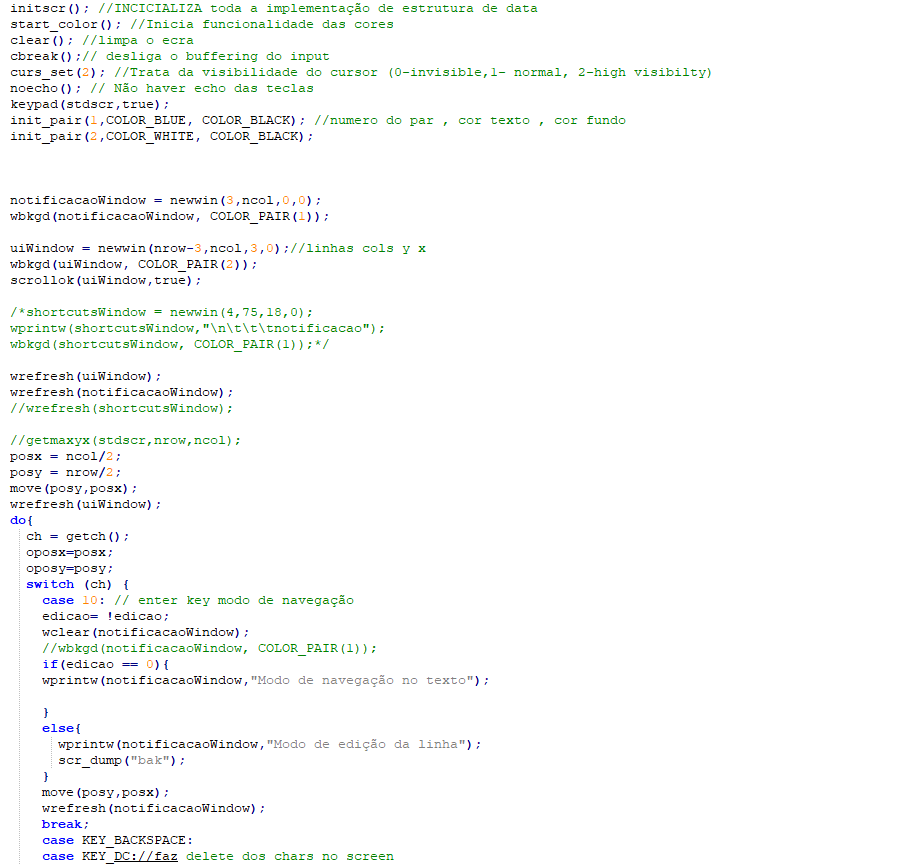
Recebe as variáveis ambiente, linhas e colunas da janela de edição (getenv). E recebe por (getopt) pela linha de comando.



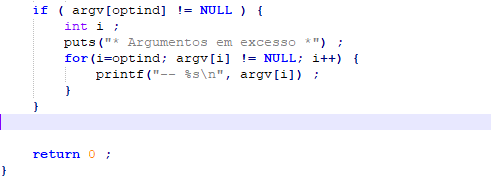
Verifica se recebeu por comando, se não recebeu pede.



Inicializa duas windows virtuais, recebe o input das direções de movimento e vários comandos, como escape, delete e enter.







**Funcionalidades Realizadas:**

Quase todas foram realizadas, com a exceção do aspell, que ficou por tratas a informação que era recebida do filho depois de ser analisada a palavra.

**Verificação e validação:**

