

Relatório *Editor de Texto*

Carlos Palma(46520) e Manuel Cunha(48482)

9 September 2021

1 Funcões

1.1 main

A função main é uma label global onde é criada a stack de forma a que todas as outras funcões tenham acesso à mesma é também o local onde ocorrem as chamadas para as outras funcões recorrendo à utilização de um switch e saltos incondicionais para que de futuro seja possível regressar às mesmas.

1.2 insert

Na função insert começamos por pedir o número da linha, que é guardado como caracter no espaço "COMAND" inicializado previamente para o respetivo efeito, guardando o carácter também num registo temporário, recorrendo à utilização de um switch verificamos em que linha pretendemos escrever fazendo uma comparação entre o registo temporário e o código ASCII correspondente à linha. Utilizando a Syscall "ReadString" pedimos ao utilizador o input desejado e de seguida apontamos o ponteiro de stack correspondente à linha em questão ao respetivo endereço da string utilizando um "store word". A função retorna à main após a escolha da linha e inserção da string.

1.3 append

Na função append começamos por pedir o número da linha, que é guardado como caracter no espaço "COMAND" inicializado previamente para o respetivo efeito, guardando o carácter também num registo temporário, recorrendo à utilização de um switch verificamos em que linha pretendemos escrever fazendo uma comparação entre o registo temporário e o código ASCII correspondente à linha. Recorrendo à função auxiliar "loop" criada por nós contamos o numero de caracteres até ao caracter nulo, guardamos o respetivo numero de caracteres num registo temporário e com o auxilio do mesmo percorremos a string até à posição do caracter nulo onde será escrita a string a adicionar pedindo ao utilizador um input recorrendo à Syscall "ReadString". A função retorna à main após a escolha da linha e concatenação da string na mesma.

1.4 change

Na função `change` começamos por pedir o número da linha, que é guardado como carácter no espaço `"COMAND"` inicializado previamente para o respetivo efeito, guardando o carácter também num registo temporário, recorrendo à utilização de um switch verificamos em que linha pretendemos escrever fazendo uma comparação entre o registo temporário e o código ASCII correspondente à linha. De seguida apagamos o texto introduzido previamente colocando um carácter nulo no início da linha e pedindo ao utilizador um input recorrendo à Syscall `"ReadString"`, guardamos o respetivo input no endereço da linha com o auxílio do ponteiro correspondente à respetiva linha. A função retorna à `main` após a escolha da linha e inserção da string sobre a linha previamente apagada.

1.5 delete

Na função `delete` começamos por pedir o número da linha, que é guardado como carácter no espaço `"COMAND"` inicializado previamente para o respetivo efeito, guardando o carácter também num registo temporário, recorrendo à utilização de um switch verificamos em que linha pretendemos escrever fazendo uma comparação entre o registo temporário e o código ASCII correspondente à linha. De seguida apagamos o texto introduzido previamente, colocando todo o conteúdo dessa linha a 0. A função retorna à `main` após a escolha da linha e reinicialização da mesma.

1.6 print

Na função `insert` começamos por pedir o número da linha, que é guardado como carácter no espaço `"COMAND"` inicializado previamente para o respetivo efeito, guardando o carácter também num registo temporário, recorrendo à utilização de um switch verificamos em que linha pretendemos escrever fazendo uma comparação entre o registo temporário e o código ASCII correspondente à linha. De seguida recorrendo a um Syscall `"PrintString"` imprimimos na tela do utilizador o output correspondente à linha previamente escolhida. Caso o utilizador insira um carácter diferente de todas as linhas existentes a função imprime como output todo o texto. A função retorna à `main` após a escolha da linha e fornecimento do output correspondente à mesma.

1.7 edit

Utilizando a Syscall `"Open"` abre o ficheiro em modo `"apenas escrita"` e copia o conteúdo do ficheiro para uma variável segura. Por fim a função retorna à `main`.

1.8 file

Através de um Syscall `"ReadString"` o utilizador introduz o nome desejado para o ficheiro e guarda-o num registo seguro. Por fim a função retorna à `main`.

1.9 write

Utilizando a Syscall "Open" abre o ficheiro em modo "apenas escrita" e copia o texto guardado no registo a0 para o ficheiro. Por fim fecha o programa através de um Syscall "Close" e retorna à main.

1.10 quit

Atraves de um Syscall guardamos o texto desejado no ficheiro e fechamos o programa

1.11 QUIT

Atraves de um Syscall fechamos o programa sem guardar no ficheiro.