

Exercício 4

Aluno: Caio Augusto Alves Nolasco

RA: 195181

Instituto de Computação

Universidade Estadual de Campinas

Campinas, 12 de Dezembro de 2020.

Sumário

1	Instruções para compilação e execução	2
2	Questão 1	2
3	Questão 2	4

1 Instruções para compilação e execução

Junto aos arquivos do código há um makefile. Para compilar o servidor e o cliente, respectivamente, basta chamar os comandos "make servidor" e "make cliente" no diretório do código. Para executar, basta chamar os executáveis criados (./servidor e ./cliente), informando ao servidor o número de porta e ao cliente, o IP do servidor e o número da porta, junto com o arquivo .txt contendo o texto de entrada redirecionado por '<', e o nome do arquivo .txt de saída redirecionado por '>'

2 Questão 1

Nas imagens a seguir, as mudanças feitas sobre o código para incorporar as mudanças em relação a chamada select() e os macros da estrutura fdset necessárias.

```
void str_cli(int sockfd){
    char *sendline, recvline[MAXLINE]; //buffers for sending and receiveing echoed message
    fd_set rset; //FD_SET struct for select()
    size_t bufsize = 4096;
    int maxfdpl, stdineof;

    stdineof = 0; //EOF flag

    sendline = (char *)malloc(bufsize * sizeof(char)); //Buffer for getline() needs to be a pointer initialized by malloc()

    FD_ZERO(&rset);

    for( ; ; ){
        if (stdineof == 0){
            FD_SET(fileno(stdin), &rset);
        }
        FD_SET(sockfd, &rset);
        //Set FD read sets

        maxfdpl = MAX(fileno(stdin), sockfd) + 1;
        //Find maximum file descriptor value plus one.

        Select(maxfdpl, &rset, NULL, NULL, NULL);
        //Call select()

        if(FD_ISSET(fileno(stdin), &rset)){
            if(Getline(&sendline, &bufsize, stdin) == -1){ //Read input file line by line.
                stdineof = 1; //If getline() == -1, found EOF.
                Shutdown(sockfd, SHUT_WR); //Call Shutdown to socket
                FD_CLR(fileno(stdin), &rset);
                continue;
            }
            Write(sockfd, sendline, strlen(sendline)); //Write to connected socket.
            bzero(sendline, bufsize);
        }
    }
}
```

```

if(FD_ISSET(fileno(stdin), &rset)){
    if(Getline(&sendline, &bufsize, stdin) == -1){ //Read input file line by line.
        stdineof = 1; //If getline() == -1, found EOF.
        Shutdown(sockfd, SHUT_WR); //Call Shutdown to socket
        FD_CLR(fileno(stdin), &rset);
        continue;
    }
    Write(sockfd, sendline, strlen(sendline)); //Write to connected socket.
    bzero(sendline, bufsize);
}

if(FD_ISSET(sockfd, &rset)){
    if(Read(sockfd, recvline, MAXLINE) == 0){ //Read echoed message
        if(stdineof == 1){
            return;
        }else{
            perror("server terminated early");
        }
    }
    fputs(recvline, stdout); //Write echoed message to stoud, output file
    bzero(recvline, sizeof(recvline));
}
}

free(sendline); //free malloc memory
}

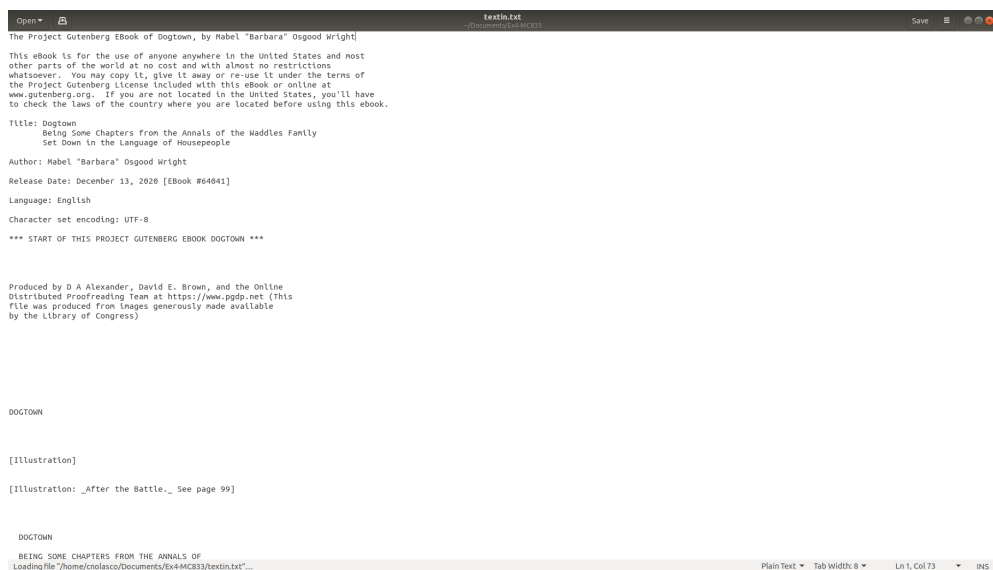
```

A chamada de cliente.c, junto com comparações entre a entrada e a saída:

```

cnolasco@calopc:~/Documents/Ex4-MC833$
cnolasco@calopc:~/Documents/Ex4-MC833$ ./cliente 127.0.0.1 33333 < textin.txt > text2out.txt
cnolasco@calopc:~/Documents/Ex4-MC833$

```



The screenshot shows a text editor window titled "textin.txt" with the following content:

The Project Gutenberg eBook of Dogtown, by Habel "Barbara" Osgood Wright

This eBook is for the use of anyone anywhere in the United States and most other parts of the world at no cost and with almost no restrictions whatsoever. You may copy it, give it away or re-use it under the terms of the Project Gutenberg license included with this eBook or online at www.gutenberg.org. If you are not located in the United States, you'll have to check the laws of the country where you are located before using this eBook.

Title: Dogtown
Being Some Chapters from the Annals of the Waddles Family
Set Down in the Language of Housepeople

Author: Habel "Barbara" Osgood Wright

Release Date: December 13, 2020 [EBook #64841]

Language: English

Character set encoding: UTF-8

*** START OF THIS PROJECT GUTENBERG EBOOK DOGTOWN ***

Produced by D A Alexander, David E. Brown, and the Online Distributed Proofreading Team at <https://www.pgdp.net> (This file was produced from images generously made available by the Library of Congress)

DOGTOWN

[Illustration]

[Illustration: _After the Battle_. See page 99]

DOGTOWN

BEING SOME CHAPTERS FROM THE ANNALS OF

Loading file "/home/cnolasco/Documents/Ex4-MC833/textin.txt"...

Plain Text ▾ Tab Width: 8 ▾ Ln 1, Col 73 ▾ INS

```

cnolasco@calopc:~/Documents/Ex4-MC833$ diff textin.txt text2out.txt
cnolasco@calopc:~/Documents/Ex4-MC833$

```

3 Questão 2

Um uso para a multiplexação de entrada e saída é que o cliente não receba o sinal de EOF caso o servidor seja encerrado enquanto o cliente executa a chamada para a leitura de texto da fonte. O uso de `select()` ou `poll()` nesse caso permite que o kernel avise o processo que seu I/O está pronto para ser processado, sem precisar que cliente fique bloqueado na chamada de leitura.