

**Московский авиационный институт
(национальный исследовательский университет)**

Факультет прикладной математики и физики

Кафедра вычислительной математики и программирования

Лабораторная работа №4 по курсу «Операционные системы»

Студент: Т.А.Габдуллин
Преподаватель: Е. С. Миронов
Группа: 80-206Б
Вариант: 9
Дата:
Оценка:
Подпись:

Лабораторная работа №4

Вариант: 9. Прочитать конкретную строку из файла

Поиск по подстроке со спец символами (/n, /r, etc). Поиск осуществлять по всему тексту

Замена фрагмента строки, на другую часть строки (возможно и пустую)

Возможность задания лимита размера файла (как минимального, так и максимального)

2 Исходный код

FileProcessor.h

```
#ifndef FILEPROCESSOR_H
#define FILEPROCESSOR_H

#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <stdbool.h>
#include <ctype.h>
#include <inttypes.h>
#include <fcntl.h>
#include <sys/types.h>
#include <sys/mman.h>
#include <sys/stat.h>
#include <unistd.h>
#include <sys/file.h>
#include <errno.h>

int Filter(int argc, char const *argv[]);
void FixSize(int *userLimit);
void StringPrint(const int fd, const int fileSize, const int RAMLimit, const int stringNumber);
void GetStringPrint(int fd, int size, char const* filename, int numStr);
void Info(const int fd, const int fileSize, const int RAMLimit);
void GetInfo(int fd, int size, char const* filename);
void CheckFileSize(int fd, char const *filename);
void Search(const int fd, const int fileSize, const int RAMLimit, char const* subString);
void GetStringSearch(int fd, int size, char const* filename, char const* subString);
char* Process(char const* src);
int RemoveOldString(const int fd, const int fileSize, const int RAMLimit, const int start, const int end);
void InsertNewString(const int fd, const int fileSize, const int RAMLimit, char const *string, const int start, const int end);
void ReplaceString(const int fd, const int fileSize, const int RAMLimit, char const *oldString, const int sizeOldStr, char const *newString, const int sizeNewStr);
void GetStringReplace(int fd, int size, char const* filename, char const* oldString, char const* newString);
```

```
void Menu(void);  
#endif
```

Main.c

```
#include <stdio.h>  
#include <string.h>  
#include <stdlib.h>  
#include <inttypes.h>  
#include <fcntl.h>  
#include <sys/types.h>  
#include <sys/stat.h>  
#include <unistd.h>  
#include <sys/io.h>  
#include <sys/mman.h>  
#include "FileProcessor.h"  
  
void menu(void);  
  
int main(int argc, char const *argv[]) {  
    int fd;  
    struct stat fileStat;  
    char fileName[256]; char newString[256]; char oldString[256];  
    int size = 0; int numberOfString;  
    int choose;  
  
    int action = Filter(argc, argv);  
    switch(action) { //Single String Mode  
        case 1:  
        {  
            GetInfo(fd, size, argv[1]); // RAM limit isn't specified by user  
            break;  
        }  
        case 2:  
        {  
            GetStringPrint(fd, size, argv[1], atoi(argv[3]));  
            break;  
        }  
        case 3:  
        {  
            GetInfo(fd, atoi(argv[2]), argv[1]);  
            break;  
        }  
        case 4:  
        {  
            GetStringPrint(fd, atoi(argv[2]), argv[1], atoi(argv[3]));  
            break;  
        }  
        case 5:  
        {  
            GetStringSearch(fd, size, argv[1], argv[3]);  
            break;  
        }  
    }  
}
```

```

    }
    case 6:
    {
        GetStringSearch(fd, atoi(argv[2]), argv[1], argv[4]);
        break;
    }
    case 7:
    {
        GetStringReplace(fd, size, argv[1], argv[3], argv[4]);
        break;
    }
    case 8:
    {
        GetStringReplace(fd, atoi(argv[2]), argv[1], argv[4], argv[5]);
        break;
    }

    case 11: // Interactive Mode
    {
        do {
            Menu();
            scanf("%d", &choise);
            switch(choise) {
                case 1:
                    printf("Input filename\n");
                    scanf("%s", fileName);
                    break;
                case 2:
                    GetInfo(fd, atoi(argv[1]), fileName);
                    break;
                case 3:
                    printf("Input number of string\n");
                    scanf("%d", &numberOfString);
                    GetStringPrint(fd, atoi(argv[1]), fileName,
numberOfString);
                    break;
                case 4:
                    printf("Input the string to replace\n");
                    scanf("%s", oldString);
                    printf("Input a new string\n");
                    memset(newString, '\0', 256);
                    scanf("%s", newString);
                    GetStringReplace(fd, atoi(argv[1]), fileName,
oldString, newString);
                    break;
                case 0:
                    break;
            }
        } while (choise!=0);
    }
}

```

```
    }  
  
    return 0;  
}
```

3 Консоль

timxag@KEKNOTE:~/Документы/lab4\$ gcc FileProcessor.c FileProcessor.h main.c

timxag@KEKNOTE:~/Документы/lab4\$./a.out

open

timxag@KEKNOTE:~/Документы/lab4\$./a.out 500

Menu:

- 1) Input the new file
- 2) Get information about the file
- 3) Print string by number
- 4) Replace the string in the file
- 0) Exit

1

Input filename

2.txt

Menu:

- 1) Input the new file
- 2) Get information about the file
- 3) Print string by number
- 4) Replace the string in the file
- 0) Exit

2

Amount of symbols - 195

Amount of lines - 5

Menu:

- 1) Input the new file
- 2) Get information about the file
- 3) Print string by number
- 4) Replace the string in the file
- 0) Exit

3

Input number of string

2

I got a song filled with shit for huy strong-willed

Menu:

- 1) Input the new file
- 2) Get information about the file
- 3) Print string by number
- 4) Replace the string in the file
- 0) Exit

4

Input the string to replace

song

Input a new string

test

Menu:

- 1) Input the new file
- 2) Get information about the file
- 3) Print string by number
- 4) Replace the string in the file
- 0) Exit

3

Input number of string

2

I got a test filled with shit for huy strong-willed

Menu:

- 1) Input the new file
- 2) Get information about the file
- 3) Print string by number
- 4) Replace the string in the file
- 0) Exit

^C

4 Вывод

Мной было изучено отображение файлов в системе Unix, написана программа на языке Си, обрабатывающая текстовые файлы — текстовый процессор.