Министерство образования Республики Беларусь

Учреждение образования

«Брестский государственный технический университет»

Кафедра ИИТ

**Лаботарорная работа №5**

**««ВВОД/ВЫВОД»**

По дисциплине «ОСиСП»

За четвертый семестр

**Выполнила:**

Студентка 2 курса

Факультета ЭИС

Группы ПО-3

Дмитрук В.Г.

**Проверила:**

Давидюк Ю.И.

**Брест 2020 г.**

**Цель работы:** изучить команды ввода и вывода.

1) Вариант 9, 19. Написать программу, которая получает со стандартного потока ввода текст руководства и выводит в стандартный поток вывода его строки, начинающиеся на гласную букву, а в поток ошибки – порядковый номер выведенной строки. Протестировать на различных

файлах с использованием конвейеров в различных комбинациях вашей программы и команд cat, sort, head, tail.

Текст программы:

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <ctype.h>

#include <fcntl.h>

#include <sys/types.h>

#include <sys/stat.h>

#include <unistd.h>

void strreverse(char\* begin, char\* end) {

char aux;

while(end>begin)

(void)(aux=\*end), (void)(\*end--=\*begin), \*begin++=aux;

}

void itoa(int value, char\* str, int base) {

static char num[] = "0123456789abcdefghijklmnopqrstuvwxyz";

char\* wstr=str;

int sign;

// Validate base

if (base<2 || base>35){ \*wstr='\0'; return; }

// Take care of sign

if ((sign=value) < 0) value = -value;

// Conversion. Number is reversed.

do \*wstr++ = num[value%base]; while(value/=base);

if(sign<0) \*wstr++='-';

\*wstr='\0';

// Reverse string

strreverse(str,wstr-1);

}

int main() {

//задание 1

char str[9000]; //сюда запишу текст руководства

char s[1000][100]; //отдельная строка текста руководства

int j=0, i;

read(0,str,9000); //считываю стандартный поток ввода

char \*line = strtok(str, "\n"); // указатель на отдельную строку

while (line!= NULL){ // пока не поделим весь str на строки

strcpy(s[j], line); // записываем новую строку

line = strtok(NULL, "\n"); // опустошаем line

j++; //переход к новой строке

}

for(i=0;i<j;i++){

int k=0;

while((s[i][k] == ' ') || (s[i][k] == '\t' )){

k++;

}

//проверяемое условие - начинается ли строка с гласной буквы

if(s[i][k] == 'A' || s[i][k] == 'E' || s[i][k] == 'Y' || s[i][k] == 'U' || s[i][k] == 'I' || s[i][k] == 'O' || s[i][k] == 'a' || s[i][k] == 'e' || s[i][k] == 'y' || s[i][k] == 'u' || s[i][k] == 'i' || s[i][k] == 'o' ) {

//запись в поток вывода

write(1,s[i],strlen(s[i]));

write(1,"\n",sizeof(char));//для того, чтобы между строками был отступ

//запись в поток ошибок порядкового номера строки

char t[25];

itoa(i,t,10);

write(2,&t,sizeof(t));

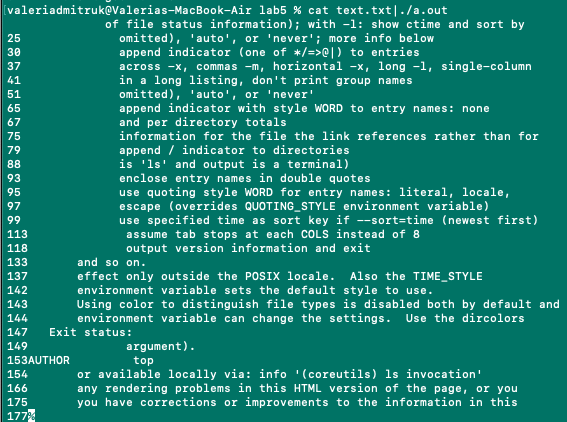
}

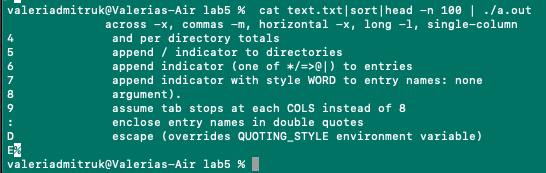
}

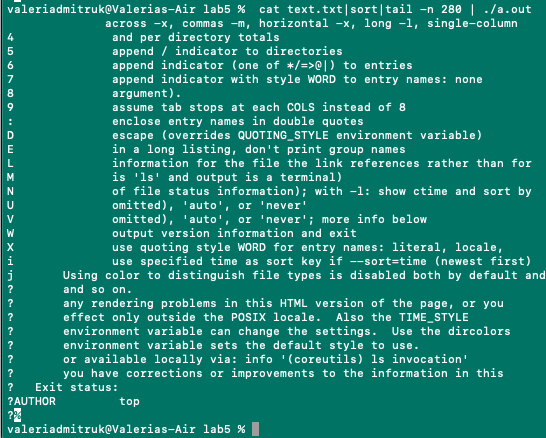
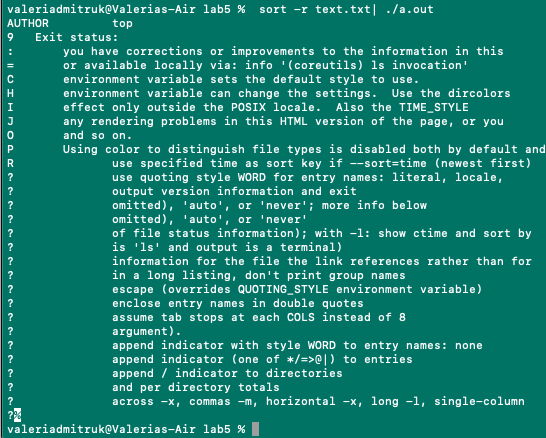
return 0;

}

Результат:





2) Ва

Вариант 9, 19. Создайте отдельные текстовые файлы для каждой секции руководства, и поместите туда эти секции.

Текст программы:

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <fcntl.h>

#include <sys/types.h>

#include <sys/stat.h>

#include <unistd.h>

int main(){

char alphabet[27] = "QWERTYUIOPASDFGHJKLZXCVBNM";//для того, чтобы потом искать секции(критерий поиска - заглавные буквы)

int p=0;

int numofsec[11]; // считает, сколько всего секций в руководстве и записывает номер строки, с которой начинается новая секция

//считываем текст руководства со стандартного потока ввода

char str[9000]; //сюда запишу текст руководства

char s[1000][100]; //отдельная строка текста руководства

int j=0, i;

read(0,str,9000); //считываю стандартный поток ввода

char \*line = strtok(str, "\n"); // указатель на отдельную строку

while (line!= NULL){ // пока не поделим весь str на строки

strcpy(s[j], line); // записываем новую строку

line = strtok(NULL, "\n"); // опустошаем line

j++; //переход к новой строке

}

//ищу конкретно названия секций руководства

for(i=0; i<j; i++)

{

int k=0;

if(strchr(alphabet, s[i][0]) != NULL) // если 1 элемент - большая буква

{

while(strchr(alphabet, s[i][k]) != NULL){ //если последующие - тоже большие буквы

printf("%c",s[i][k]); //просмотр наименования секции (посимвольно)

k++;

}

numofsec[p] = i; //запоминаем начало секции

printf("%d",numofsec[p]); // выводим номер строки, с которой начинается секция

p++;

printf("\n");

}

}

numofsec[10] = j;

printf("%d",numofsec[p]);

p++;

printf("\n");

char filename[10000]; //имя создаваемого файла

char num[1000];

//создание отдельного файла для каждой секции

for(int i=0; i<p-1;i++){

//создание отдельного имени для каждого файла

char t = i + '0';

char txt[] = ".txt";

strcpy(num, "");

strcat(num, &t);

strcpy(filename, "");

strcat(filename, num);

strcat(filename,txt);

printf("%s",filename);

int myfile = open(filename, O\_RDWR| O\_CREAT | O\_TRUNC,0666); //открытие файла

if (myfile<0)

{ //если не могу открыть (создать) файл

printf("Can't open file with name %s", filename);

exit(-1);

}

else

{

for(int h=numofsec[i]; h<numofsec[i+1]; h++){

printf("%d",h); // вывожу номер строки

printf("\n");

printf("%s",s[h]); // вывожу руководство

write(myfile, s[h], strlen(s[h])); //запись в файл

}

}

close(myfile); //закрываю файл

if (myfile<0)

{//если не могу закрыть файл

printf("Can't close file with name %s", filename);

exit(-1);

}

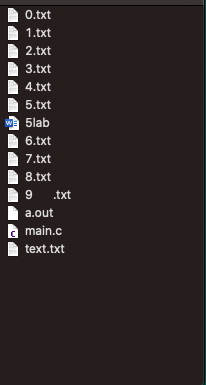
}

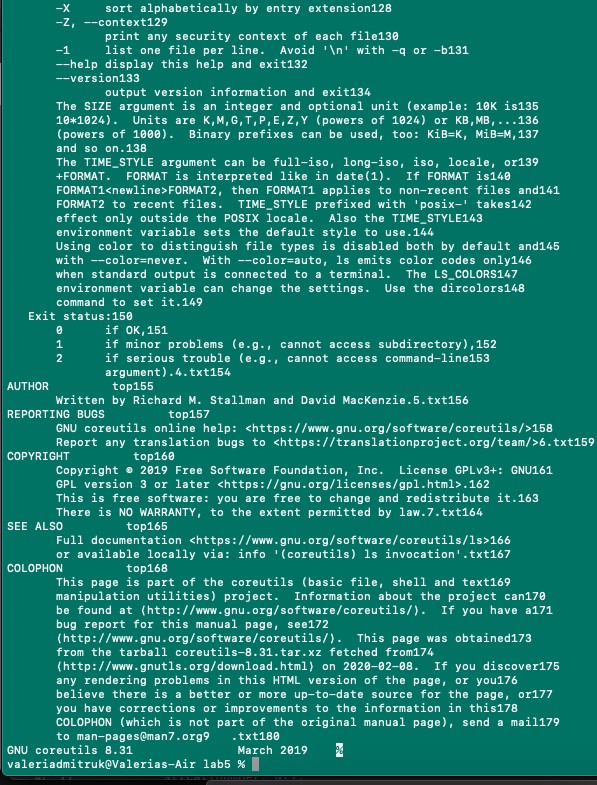
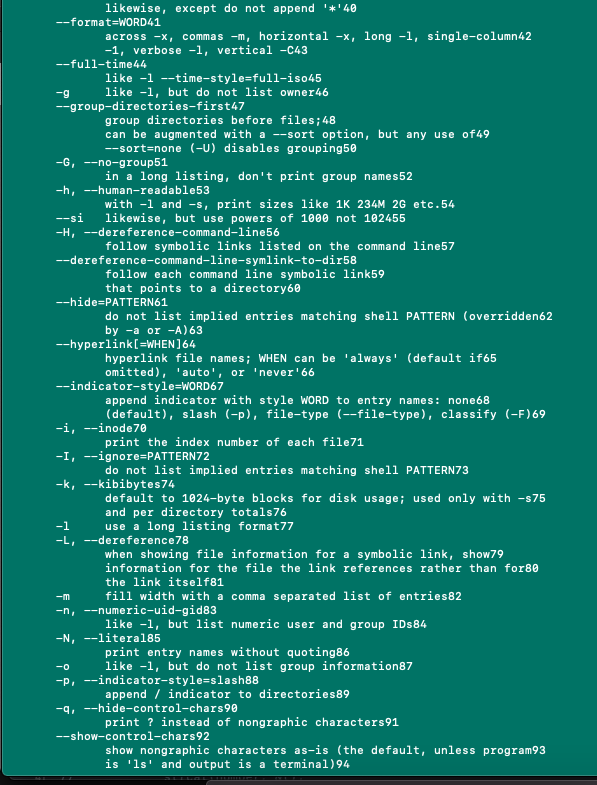
return 0;

}

Результат:

(созданные файлы находятся в репозитории на github)





Вывод: Изучила команды ввода и вывода.