

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:

```

[valeriadmitruk@Valerias-MacBook-Air lab5 % cat text.txt|./a.out
of file status information); with -l: show ctime and sort by
25      omitted), 'auto', or 'never'; more info below
30      append indicator (one of */=>@|) to entries
37      across -x, commas -m, horizontal -x, long -l, single-column
41      in a long listing, don't print group names
51      omitted), 'auto', or 'never'
65      append indicator with style WORD to entry names: none
67      and per directory totals
75      information for the file the link references rather than for
79      append / indicator to directories
88      is 'ls' and output is a terminal)
93      enclose entry names in double quotes
95      use quoting style WORD for entry names: literal, locale,
97      escape (overrides QUOTING_STYLE environment variable)
99      use specified time as sort key if --sort=time (newest first)
113     assume tab stops at each COLS instead of 8
118     output version information and exit
133     and so on.
137     effect only outside the POSIX locale. Also the TIME_STYLE
142     environment variable sets the default style to use.
143     Using color to distinguish file types is disabled both by default and
144     environment variable can change the settings. Use the dircolors
147     Exit status:
149         argument).
153AUTHOR      top
154     or available locally via: info '(coreutils) ls invocation'
166     any rendering problems in this HTML version of the page, or you
175     you have corrections or improvements to the information in this
177%

```

```
valeriadmitruk@Valerias-Air lab5 % cat text.txt|sort|head -n 100 | ./a.out
        across -x, commas -m, horizontal -x, long -l, single-column
4        and per directory totals
5        append / indicator to directories
6        append indicator (one of */=>@|) to entries
7        append indicator with style WORD to entry names: none
8        argument).
9        assume tab stops at each COLS instead of 8
:        enclose entry names in double quotes
D        escape (overrides QUOTING_STYLE environment variable)
E%
valeriadmitruk@Valerias-Air lab5 %
```

```
valeriadmitruk@Valerias-Air lab5 % sort -r text.txt| ./a.out
AUTHOR      top
9  Exit status:
:    you have corrections or improvements to the information in this
=    or available locally via: info '(coreutils) ls invocation'
C    environment variable sets the default style to use.
H    environment variable can change the settings. Use the dircolors
I    effect only outside the POSIX locale. Also the TIME_STYLE
J    any rendering problems in this HTML version of the page, or you
O    and so on.
P    Using color to distinguish file types is disabled both by default and
R    use specified time as sort key if --sort=time (newest first)
?    use quoting style WORD for entry names: literal, locale,
?    output version information and exit
?    omitted), 'auto', or 'never'; more info below
?    omitted), 'auto', or 'never'
?    of file status information); with -l: show ctime and sort by
?    is 'ls' and output is a terminal)
?    information for the file the link references rather than for
?    in a long listing, don't print group names
?    escape (overrides QUOTING_STYLE environment variable)
?    enclose entry names in double quotes
?    assume tab stops at each COLS instead of 8
?    argument).
?    append indicator with style WORD to entry names: none
?    append indicator (one of */=>@|) to entries
?    append / indicator to directories
?    and per directory totals
?    across -x, commas -m, horizontal -x, long -l, single-column
?%
valeriadmitruk@Valerias-Air lab5 %
```

```

valeriadmitruk@Valerias-Air lab5 % cat text.txt|sort|tail -n 280 | ./a.out
        across -x, commas -m, horizontal -x, long -l, single-column
4         and per directory totals
5         append / indicator to directories
6         append indicator (one of */=>@|) to entries
7         append indicator with style WORD to entry names: none
8         argument).
9         assume tab stops at each COLS instead of 8
:        enclose entry names in double quotes
D        escape (overrides QUOTING_STYLE environment variable)
E        in a long listing, don't print group names
L        information for the file the link references rather than for
M        is 'ls' and output is a terminal)
N        of file status information); with -l: show ctime and sort by
U        omitted), 'auto', or 'never'
V        omitted), 'auto', or 'never'; more info below
W        output version information and exit
X        use quoting style WORD for entry names: literal, locale,
i        use specified time as sort key if --sort=time (newest first)
j        Using color to distinguish file types is disabled both by default and
?        and so on.
?        any rendering problems in this HTML version of the page, or you
?        effect only outside the POSIX locale. Also the TIME_STYLE
?        environment variable can change the settings. Use the dircolors
?        environment variable sets the default style to use.
?        or available locally via: info '(coreutils) ls invocation'
?        you have corrections or improvements to the information in this
?        Exit status:
?AUTHOR      top
?%
valeriadmitruk@Valerias-Air lab5 %

```

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
    {

```

```

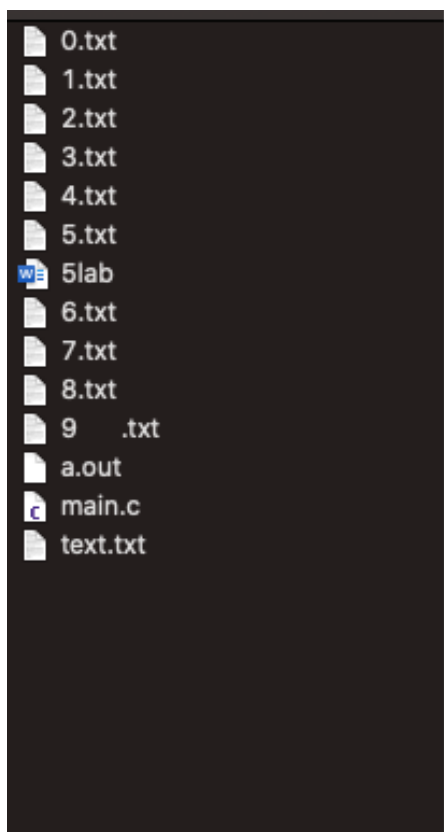
        for(int h=numofsec[i]; h<numofsec[i+1]; h++){
            printf("%d",h); // вывожу номер строки
            printf("\n");
            printf("%s",s[h]); // вывожу руководство
            write(myfile, s[h], strlen(s[h])); //запись в файл
        }
    }
    close(myfile); //закрываю файл

    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)




```

valeriadmitruk@Valerias-Air lab5 % gcc main.c
valeriadmitruk@Valerias-Air lab5 % cat text.txt|./a.out
LS0
NAME1
SYNOPSIS3
DESCRIPTION5
AUTHOR154
REPORTING156
COPYRIGHT159
SEE164
COLOPHON167
GNU180
181
0.txt0
LS(1)                                User Commands                                LS(1)1.txt1
NAME                                top2
    ls - list directory contents2.txt3
SYNOPSIS                            top4
    ls [OPTION]... [FILE]...3.txt5
DESCRIPTION                        top6
    List information about the FILES (the current directory by default).7
    Sort entries alphabetically if none of -cftuvSUX nor --sort is8
    specified.9
    Mandatory arguments to long options are mandatory for short options10
    too.11
    -a, --all12
        do not ignore entries starting with .13
    -A, --almost-all14
        do not list implied . and ..15
    --author16
        with -l, print the author of each file17
    -b, --escape18
        print C-style escapes for nongraphic characters19
    --block-size=SIZE20
        with -l, scale sizes by SIZE when printing them; e.g.,21
        '--block-size=M'; see SIZE format below22
    -B, --ignore-backups23
        do not list implied entries ending with ~24
    -c        with -lt: sort by, and show, ctime (time of last modification25
        of file status information); with -l: show ctime and sort by26
        name; otherwise: sort by ctime, newest first27
    -C        list entries by columns28
    --color[=WHEN]29
        colorize the output; WHEN can be 'always' (default if30
        omitted), 'auto', or 'never'; more info below31
    -d, --directory32
        list directories themselves, not their contents33
    -D, --dired34
        generate output designed for Emacs' dired mode35
    -f        do not sort, enable -aU, disable -ls --color36
    -F, --classify37
        append indicator (one of */=>@|) to entries38
    --file-type39
        likewise, except do not append '*'40
    --format=WORD41

```

```

    likewise, except do not append '*'40
--format=WORD41
    across -x, commas -m, horizontal -x, long -l, single-column42
    -l, verbose -l, vertical -C43
--full-time44
    like -l --time-style=full-iso45
-g    like -l, but do not list owner46
--group-directories-first47
    group directories before files;48
    can be augmented with a --sort option, but any use of49
    --sort=none (-U) disables grouping50
-G, --no-group51
    in a long listing, don't print group names52
-h, --human-readable53
    with -l and -s, print sizes like 1K 234M 2G etc.54
--si    likewise, but use powers of 1000 not 102455
-H, --dereference-command-line56
    follow symbolic links listed on the command line57
--dereference-command-line-symlink-to-dir58
    follow each command line symbolic link59
    that points to a directory60
--hide=PATTERN61
    do not list implied entries matching shell PATTERN (overridden62
    by -a or -A)63
--hyperlink[=WHEN]64
    hyperlink file names; WHEN can be 'always' (default if65
    omitted), 'auto', or 'never'66
--indicator-style=WORD67
    append indicator with style WORD to entry names: none68
    (default), slash (-p), file-type (--file-type), classify (-F)69
-i, --inode70
    print the index number of each file71
-I, --ignore=PATTERN72
    do not list implied entries matching shell PATTERN73
-k, --kibibytes74
    default to 1024-byte blocks for disk usage; used only with -s75
    and per directory totals76
-l    use a long listing format77
-L, --dereference78
    when showing file information for a symbolic link, show79
    information for the file the link references rather than for80
    the link itself81
-m    fill width with a comma separated list of entries82
-n, --numeric-uid-gid83
    like -l, but list numeric user and group IDs84
-N, --literal85
    print entry names without quoting86
-o    like -l, but do not list group information87
-p, --indicator-style=slash88
    append / indicator to directories89
-q, --hide-control-chars90
    print ? instead of nongraphic characters91
--show-control-chars92
    show nongraphic characters as-is (the default, unless program93
    is 'ls' and output is a terminal)94

```

```

--show-control-chars92
    show nongraphic characters as-is (the default, unless program93
    is 'ls' and output is a terminal)94
-Q, --quote-name95
    enclose entry names in double quotes96
--quoting-style=WORD97
    use quoting style WORD for entry names: literal, locale,98
    shell, shell-always, shell-escape, shell-escape-always, c,99
    escape (overrides QUOTING_STYLE environment variable)100
-r, --reverse101
    reverse order while sorting102
-R, --recursive103
    list subdirectories recursively104
-s, --size105
    print the allocated size of each file, in blocks106
-S
    sort by file size, largest first107
--sort=WORD108
    sort by WORD instead of name: none (-U), size (-S), time (-t),109
    version (-v), extension (-X)110
--time=WORD111
    with -l, show time as WORD instead of default modification112
    time: atime or access or use (-u); ctime or status (-c); also113
    use specified time as sort key if --sort=time (newest first)114
--time-style=TIME_STYLE115
    time/date format with -l; see TIME_STYLE below116
-t
    sort by modification time, newest first117
-T, --tabsize=COLS118
    assume tab stops at each COLS instead of 8119
-u
    with -lt: sort by, and show, access time; with -l: show access120
    time and sort by name; otherwise: sort by access time, newest121
    first122
-U
    do not sort; list entries in directory order123
-v
    natural sort of (version) numbers within text124
-w, --width=COLS125
    set output width to COLS. 0 means no limit126
-x
    list entries by lines instead of by columns127
-X
    sort alphabetically by entry extension128
-Z, --context129
    print any security context of each file130
-l
    list one file per line. Avoid '\n' with -q or -b131
--help display this help and exit132
--version133
    output version information and exit134
The SIZE argument is an integer and optional unit (example: 10K is135
10*1024). Units are K,M,G,T,P,E,Z,Y (powers of 1024) or KB,MB,...136
(powers of 1000). Binary prefixes can be used, too: KiB=K, MiB=M,137
and so on.138
The TIME_STYLE argument can be full-iso, long-iso, iso, locale, or139
+FORMAT. FORMAT is interpreted like in date(1). If FORMAT is140
FORMAT1<newline>FORMAT2, then FORMAT1 applies to non-recent files and141
FORMAT2 to recent files. TIME_STYLE prefixed with 'posix-' takes142
effect only outside the POSIX locale. Also the TIME_STYLE143
environment variable sets the default style to use.144
Using color to distinguish file types is disabled both by default and145
with --color=never. With --color=auto, ls emits color codes only146

```



```

-X      sort alphabetically by entry extension128
-Z, --context129
        print any security context of each file130
-1      list one file per line.  Avoid '\n' with -q or -b131
--help  display this help and exit132
--version133
        output version information and exit134
The SIZE argument is an integer and optional unit (example: 10K is135
10*1024).  Units are K,M,G,T,P,E,Z,Y (powers of 1024) or KB,MB,...136
(powers of 1000).  Binary prefixes can be used, too: KiB=K, MiB=M,137
and so on.138
The TIME_STYLE argument can be full-iso, long-iso, iso, locale, or139
+FORMAT.  FORMAT is interpreted like in date(1).  If FORMAT is140
FORMAT1<newline>FORMAT2, then FORMAT1 applies to non-recent files and141
FORMAT2 to recent files.  TIME_STYLE prefixed with 'posix-' takes142
effect only outside the POSIX locale.  Also the TIME_STYLE143
environment variable sets the default style to use.144
Using color to distinguish file types is disabled both by default and145
with --color=never.  With --color=auto, ls emits color codes only146
when standard output is connected to a terminal.  The LS_COLORS147
environment variable can change the settings.  Use the dircolors148
command to set it.149
Exit status:150
  0      if OK,151
  1      if minor problems (e.g., cannot access subdirectory),152
  2      if serious trouble (e.g., cannot access command-line153
        argument).4.txt154
AUTHOR      top155
        Written by Richard M. Stallman and David MacKenzie.5.txt156
REPORTING BUGS      top157
        GNU coreutils online help: <https://www.gnu.org/software/coreutils/>158
        Report any translation bugs to <https://translationproject.org/team/>6.txt159
COPYRIGHT      top160
        Copyright © 2019 Free Software Foundation, Inc.  License GPLv3+: GNU161
        GPL version 3 or later <https://gnu.org/licenses/gpl.html>.162
        This is free software: you are free to change and redistribute it.163
        There is NO WARRANTY, to the extent permitted by law.7.txt164
SEE ALSO      top165
        Full documentation <https://www.gnu.org/software/coreutils/ls>166
        or available locally via: info '(coreutils) ls invocation'.txt167
COLOPHON      top168
        This page is part of the coreutils (basic file, shell and text169
        manipulation utilities) project.  Information about the project can170
        be found at (http://www.gnu.org/software/coreutils/).  If you have a171
        bug report for this manual page, see172
        (http://www.gnu.org/software/coreutils/).  This page was obtained173
        from the tarball coreutils-8.31.tar.xz fetched from174
        (http://www.gnutls.org/download.html) on 2020-02-08.  If you discover175
        any rendering problems in this HTML version of the page, or you176
        believe there is a better or more up-to-date source for the page, or177
        you have corrections or improvements to the information in this178
        COLOPHON (which is not part of the original manual page), send a mail179
        to man-pages@man7.org9 .txt180
GNU coreutils 8.31      March 2019      %
valeriadmitruk@Valerias-Air lab5 %

```

Вывод: Изучила команды ввода и вывода.