

Министерство образования Республики Беларусь
Учреждение образования
«Брестский государственный технический университет»
Кафедра ИИТ

Лабораторная работа №5
По дисциплине: «Операционные системы и системное программирование»
Тема: «Ввод/вывод»

Выполнил
Студент 2 курса
Группы ПО-3
Новикович А.А.
Проверил:
Давидюк Ю.И.

Брест 2020

Лабораторная работа №5

Вариант 18.

Задания:

1. 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.
2. В текущем каталоге создайте два файла, в один из которых выводите UID и имя пользователя, а в другой – GID и имя пользователя.

Код программы:

```
#include <fcntl.h>
#include <stdio.h>
#include <string.h>
#include <sys/types.h>
#include <unistd.h>
#include <stdlib.h>
#include <locale.h>

void lab(char *f){
char buf[512][150] = {}; ;
int size[180];
int clone[512][50];
int j = 0, i = 0, q = 0;
while(f[q] != '\0')
{
if(f[q] == '\n')
{
size[i] = j;
j = 0; q++; i++;
}
else
{
buf[i][j] = f[q];
j++; q++;
}
}

int k = 0; int index = i; int l = 0; int tmp[180];
char kkk[255][15];
for (int i = 0; i < index; i++)
{
for (int j = 0; j < 170; j++)
{
```

```

if(buf[i][j] == ':')
{
k++;
}
if(k == 2)
{
if(buf[i][j] != ':')
{
clone[i][l] = (buf[i][j] - '0');
kkk[i][l] = buf[i][j];
l++;
}
}
if(k == 3)
{
tmp[i] = l - 1;
i++;
j = 0;
l = 0;
k = 0;
}
}
}
char chet[255]; int temp = 0;
for(int p = 0; p < index; p++)
{
if(clone[p][tmp[p]] % 2 == 0)
{
for (int i = 0; i < tmp[p] + 1; i++)
{
chet[temp] = kkk[p][i]; temp++;
}
chet[temp] = ',';
temp++;
}
}

char ne_chet[255]; int temp1 = 0;
for(int p = 0; p < index; p++)
{
if(clone[p][tmp[p]] % 2 != 0)
{
for (int i = 0; i < tmp[p] + 1; i++)
{
ne_chet[temp1] = kkk[p][i]; temp1++;
}
ne_chet[temp1] = ',';
temp1++;
}
}
printf("-----\n");

```

```
printf("Поток ввода - чётные\n");
write(1, chet, temp);
printf("\n");
printf("-----\n");
printf("Поток ошибок - нечётные\n");
write(2, ne_chet, temp1);
printf("\n");
```

```
int top1[180];
char gid[255][15];
for (int i = 0; i < index; i++)
{
    for (int j = 0; j < 170; j++)
    {
        if(buf[i][j] == ':')
        {
            k++;
        }
        if(k == 3)
        {
            if(buf[i][j] != ':')
            {
                gid[i][l] = buf[i][j];
                l++;
            }
        }
        if(k == 4)
        {
            top1[i] = l - 1;
            i++;
            j = 0;
            l = 0;
            k = 0;
        }
    }
}
```

```
int top[180];
char name[255][15];
for (int i = 0; i < index; i++)
{
    for (int j = 0; j < 170; j++)
    {
        if(buf[i][j] == ':')
        {
            k++;
        }
        if(k == 0)
        {
            if(buf[i][j] != ':')
            {
```

```

name[i][l] = buf[i][j];
l++;
}
}
if(k == 1)
{
top[i] = l - 1 ;
i++;
j = -1;
l = 0;
k = 0;
}
}
}
int a = open("file1", O_RDWR | O_TRUNC | O_CREAT, 0777);
int b = open("file2", O_RDWR | O_TRUNC | O_CREAT, 0777);
char file1[170][50] = {};
char file2[170][50] = {};

printf("-----\n");

printf("name + GID \n");
for(int p = 0; p < index; p++) // GID + NAME
{
int k=0;
for(int j = 0; j < top[p] + 1; j++)
{
file1[p][j] = name[p][j];
k++;
}
file1[p][k] = ' ';
k++;

for(int j = 0; j < top1[p] + 1; j++)
{
file1[p][k] = gid[p][j];
k++;
}
}

for(int p = 0; p < index; p++)
{
for(int j = 0; j < top[p] + top1[p] + 3; j++)
{
printf("%c", file1[p][j]);
}
printf("\n");
}
printf("-----\n");
printf("name + UID \n");

```

```

for(int p = 0; p < index; p++) // UID + name
{
int k = 0;
for(int j = 0; j < top[p] + 1; j++)
{
file2[p][j] = name[p][j];
k++;
}
file2[p][k] = ' ';
k++;

```

```

for(int j = 0; j < top1[p] + 1; j++)
{
file2[p][k] = kkk[p][j];
k++;
}
}

```

```

for(int p = 0; p < index; p++)
{
for(int j = 0; j < top[p] + top1[p] + 3; j++)
{
printf("%c", file2[p][j]);
}
printf("\n");
}
write(b,file2,256);
write(a,file1,256);
close(a);
close(b);
}

```

```

int main(){
setlocale (LC_ALL, "ru");
char f[2500];
int size;
if((size = read(0,f,2500)) < 0)
{
printf("Ошибка %d\n", size);
exit(1);
}
else
{
printf("Удачно %d\n", size);
}
laba(f);
printf("\n");
return 0;
}

```

Задание 1.

```
artyom@AN:~/OS$ cat /etc/passwd | ./a.out
```

Удачно 2484

Поток ввода - чётные

0,2,4,6,8,10,34,38,65534,100,102,104,106,108,110,112,114,116,118,120,1000,122,

Поток ошибок - нечётные

1,3,5,7,9,13,33,39,41,101,103,105,107,109,111,113,115,117,119,121,

name + GID

root 0

daemon 1

bin 2

sys 3

sync 65534

games 60

name + UID

root 0

daemon 1

bin 2

sys 3

sync 4

games 5

man 6

lp 7

mail 8

news 9

uucp 10

```
artyom@AN:~/OS$ cat /etc/passwd | sort -r | ./a.out
```

Удачно 2484

Поток ввода - чётные

112,10,100,102,4,114,0,122,65534,6,8,38,118,120,108,110,2,34,116,106,1000,104,

Поток ошибок - нечётные

33,105,107,3,101,111,109,115,13,9,103,7,113,39,41,119,121,5,1,117,

name + GID

www-data 33

whoopsie 117

uudd 111

uucp 10

usbmux 46

sys 3

systemd-resolve 103

systemd-network 102

syslog 106

sync 65534

```
name + UID
www-data 33
whoopsie 112
uidd 105
uucp 10
usbmux 10
sys 3
systemd-resolve 101
systemd-network 100
syslog 102
sync 4
speech-dispatcher 11
saned 114
rtkit 109
```

Вывод: изучил работу с системными вызовами.