

Отчёт по лабораторной работе

Дисциплина: Архитектура компьютера

Мутале Чали

Содержание

1	Цель работы	1
2	Выполнение лабораторной работы	1
3	Выполнение самостоятельной работы.....	4
4	Выводы.....	7
5	Список литературы.....	7

1 Цель работы

Приобретение навыков написания программ для работы с файлами.

2 Выполнение лабораторной работы

Создаю каталог для программ лабораторной работы № 10, перехожу в него и создаю файлы lab10-1.asm, readme-1.txt и readme-2.txt:

```
(cmutale@cmutale)-[~]
$ mkdir -p ~/work/arch-pc/lab10

(cmutale@cmutale)-[~]
$ cd ~/work/arch-pc/lab10

(cmutale@cmutale)-[~/work/arch-pc/lab10]
$ touch lab10-1.asm readme-1.txt readme-2.txt

(cmutale@cmutale)-[~/work/arch-pc/lab10]
$ ls
lab10-1.asm  readme-1.txt  readme-2.txt

(cmutale@cmutale)-[~/work/arch-pc/lab10]
$ cp ~/Downloads/in_out.asm in_out.asm
```

рис 1

Ввожу в файл lab10-1.asm текст программы (Программа записи в файл сообщения):

```
1 %include 'in_out.asm'
2
3 SECTION .data
4 filename db 'readme-1.txt', 0h
5 msg db 'Введите строку для записи в файл: ', 0h ; Сообщение
6
7 SECTION .bss
8 contents resb 255 ; переменная для вводимой строки
9 SECTION .text
10 global _start
11 _start:
12
13 ; --- Печать сообщения `msg`
14 mov eax, msg
15 call sprint
16
17 ; --- Запись введенной с клавиатуры строки в `contents`
18 mov ecx, contents
19 mov edx, 255
20 call sread
21
22 ; --- Открытие существующего файла (`sys_open`)
23 mov ecx, 2 ; открываем для записи (2)
24 mov ebx, filename
25 mov eax, 5
26 int 80h
27
28 mov esi, eax
29
30 ; --- Расчет длины введенной строки
31 mov eax, contents ; в `eax` запишется количество
32 call slen ; введенных байтов
33
34 ; --- Записываем в файл `contents` (`sys_write`)
35 mov edx, eax
36 mov ecx, contents
37 mov ebx, esi
38 mov eax, 4
39 int 80h
40
41 ; --- Закрываем файл (`sys_close`)
42 mov ebx, esi
43 mov eax, 6
44 int 80h
45 call quit
```

рис 2

Создаю исполняемый файл и проверяю его работу:

```
(cmutale@cmutale)-[~/work/arch-pc/lab10]
$ nasm -f elf lab10-1.asm

(cmutale@cmutale)-[~/work/arch-pc/lab10]
$ ld -m elf_i386 -o lab10-1 lab10-1.o

(cmutale@cmutale)-[~/work/arch-pc/lab10]
$ ./lab10-1
Введите строку для записи в файл: Hello. My name is Chali
```

рис 3

```
(cmutale@cmutale)-[~/work/arch-pc/lab10]
$ cat readme-1.txt
Hello. My name is Chali
```

рис 4

С помощью `chmod` изменяю права доступа к файлу `lab10-1`, запретив его выполнение(`u-x`). При запуске, получаю сообщение “Permission denied” (отказано в разрешении):

```
(cmutale@cmutale)-[~/work/arch-pc/lab10]
$ chmod u-x lab10-1

(cmutale@cmutale)-[~/work/arch-pc/lab10]
$ ./lab10-1
zsh: permission denied: ./lab10-1
```

рис 5

С помощью `chmod` изменяю права доступа к файлу `lab10-1.asm` с исходным текстом программы, добавляя права на исполнение(`u+x`), и проверяю с помощью `ls -l`:

```
(cmutale@cmutale)-[~/work/arch-pc/lab10]
$ chmod u+x lab10-1.asm

(cmutale@cmutale)-[~/work/arch-pc/lab10]
$ ls -l lab10-1.asm
-rwxr--r-- 1 cmutale cmutale 1123 Dec 14 11:35 lab10-1.asm
```

рис 6

При исполнении, программа изменяет доступ к файлу `lab10-1`. При запуске, программа добавляет введенный текст к старому введенному тексту, именно поэтому мы видим две строки:

```
(cmutale@cmutale)-[~/work/arch-pc/lab10]
$ nasm -f elf -g -l lab10-1.lst lab10-1.asm

(cmutale@cmutale)-[~/work/arch-pc/lab10]
$ ld -m elf_i386 -o lab10-1 lab10-1.o

(cmutale@cmutale)-[~/work/arch-pc/lab10]
$ ./lab10-1
Введите строку для записи в файл: Hello

(cmutale@cmutale)-[~/work/arch-pc/lab10]
$ cat readme-1.txt
Hello
My name is Chali
```

рис 7

В соответствии с вариантом 10, устанавливаю права доступа к файлу readme1.txt в символьном виде(r- r- rwx), а для файла readme-2.txt – в восьмеричном виде(447). Проверяю правильность выполнения с помощью команды ls -l:

```
(cmutale@cmutale)-[~/work/arch-pc/lab10]
$ chmod u+r--,o+rwx readme-1.txt

(cmutale@cmutale)-[~/work/arch-pc/lab10]
$ chmod 447 readme-2.txt

(cmutale@cmutale)-[~/work/arch-pc/lab10]
$ ls -l
total 44
-rw-r--r-- 1 cmutale cmutale 3480 Dec 14 11:13 in_out.asm
-rwxr-xr-x 1 cmutale cmutale 9736 Dec 14 11:41 lab10-1
-rwxr--r-- 1 cmutale cmutale 1123 Dec 14 11:35 lab10-1.asm
-rw-r--r-- 1 cmutale cmutale 13045 Dec 14 11:40 lab10-1.lst
-rw-r--r-- 1 cmutale cmutale 2512 Dec 14 11:40 lab10-1.o
-rw-r--rwx 1 cmutale cmutale 24 Dec 14 11:41 readme-1.txt
-r--r--rwx 1 cmutale cmutale 0 Dec 14 11:11 readme-2.txt
```

рис 8

3 Выполнение самостоятельной работы

Создаю файл task10.asm и в него пишу текст программы:

```
(cmutale@cmutale)-[~/work/arch-pc/lab10]
$ touch task10.asm

(cmutale@cmutale)-[~/work/arch-pc/lab10]
$ gedit task10.asm
```

рис 9

```

1 %include 'in_out.asm'
2
3 SECTION .data
4 one: db 'Как Вас зовут?',0
5 filename: db "task10.txt", 0
6 msg: db 'Меня зовут '
7 theLength: equ $-msg
8
9 SECTION .bss
10 name: resb 64
11
12 SECTION .text
13 global _start
14
15 _start:
16 mov eax,one
17 call sprintfLF
18
19 mov ecx, name
20 mov edx, 64
21 call sread
22
23 mov ecx, 0777o
24 mov ebx, filename
25 mov eax, 8
26 int 80h
27
28 mov ecx, 2
29 mov ebx, filename
30 mov eax, 5
31 int 80h
32
33 mov eax, 4
34 mov ebx,eax
35 mov ecx,msg
36 int 80h
37
38 mov ebx,esi
39 mov eax,6
40 int 80h
41 call quit
42

```

рис 10

Создаю исполняемый файл. Программа выводит приглашения “Как Вас зовут?”, вводится с клавиатуры свои фамилию и имя, создает файл с именем name.txt, записывает в файл сообщение “Меня зовут”, допишет в файл строку введенную с клавиатуры и закрывает файл:

```

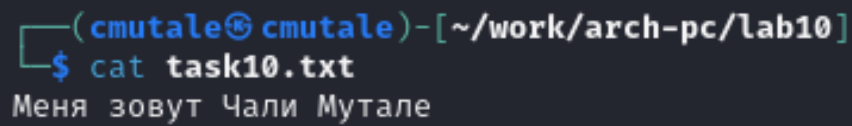
(cmutale@cmutale)-[~/work/arch-pc/lab10]
$ nasm -f elf -g -l task10.lst task10.asm

(cmutale@cmutale)-[~/work/arch-pc/lab10]
$ ld -m elf_i386 -o task10 task10.o

(cmutale@cmutale)-[~/work/arch-pc/lab10]
$ ./task10
Как Вас зовут?
Чали Мутале

```

рис 11



```
(cmutale@cmutale)-[~/work/arch-pc/lab10]  
$ cat task10.txt  
Меня зовут Чали Мутале
```

рис 12

Код программы:

```
%include 'in_out.asm'
```

```
SECTION .data  
one: db 'Как Вас зовут?',0  
filename: db "name.txt", 0  
msg: db 'Меня зовут '  
theLength: equ $-msg
```

```
SECTION .bss  
name: resb 64
```

```
SECTION .text  
global _start
```

```
_start:  
mov eax,one  
call sprintLF
```

```
mov ecx, name  
mov edx, 64  
call sread
```

```
mov ecx, 0777o  
mov ebx, filename  
mov eax, 8  
int 80h
```

```
mov ecx, 2  
mov ebx, filename  
mov eax, 5  
int 80h
```

```
mov eax, 4  
mov ebx,eax  
mov ecx,msg  
int 80h
```

```
mov ebx,esi  
mov eax,6  
int 80h  
call quit
```

4 Выводы

При выполнении лабораторной работы, я приобрела навыки написания программ для работы с файлами.

5 Список литературы

Архитектура ЭВМ