

Отчёт по лабораторной работе 10

дисциплина: Архитектура компьютера

Симдянов Тимур НПИбд-03-24

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
2.1	Задание для самостоятельной работы	9
3	Выводы	12

Список иллюстраций

2.1	Исходный код программы lab10-1.asm	7
2.2	Запуск программы lab10-1.asm	8
2.3	Запрещенный запуск файла	8
2.4	Файл lab10-1.asm с правами на выполнение	9
2.5	Настройка прав доступа к файлам	9
2.6	Исходный код программы lab10-2.asm	10
2.7	Результат выполнения программы lab10-2.asm	11

Список таблиц

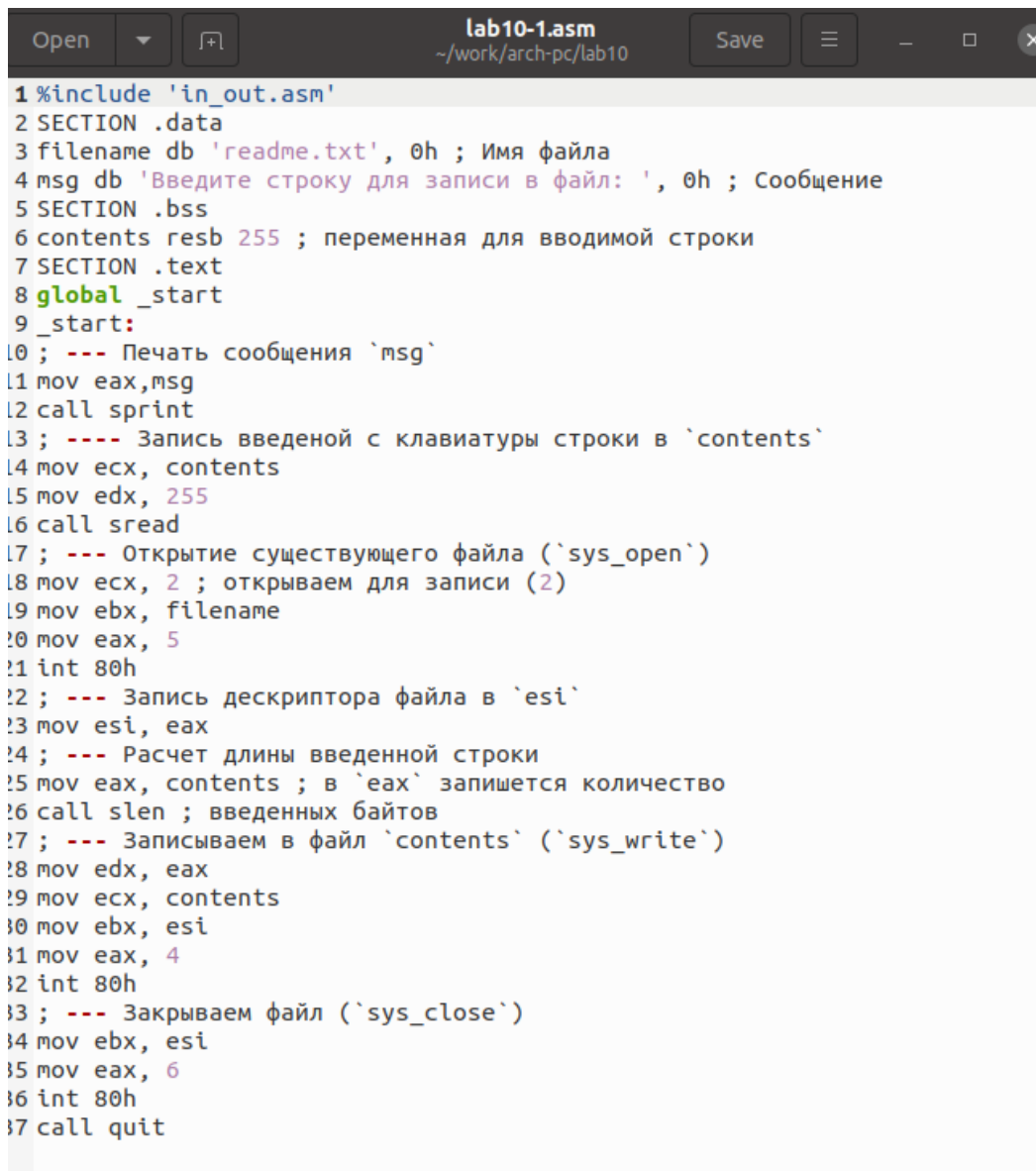
1 Цель работы

Целью работы является приобретение навыков написания программ для работы с файлами.

2 Выполнение лабораторной работы

Создал каталог для программ лабораторной работы №10, перешел в него и создал файлы lab10-1.asm, readme-1.txt и readme-2.txt.

В файл lab10-1.asm добавил текст программы из листинга 10.1, которая выполняет запись сообщения в файл. Скомпилировал программу, создал исполняемый файл и проверил её функциональность.



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

Рис. 2.1: Исходный код программы lab10-1.asm

Программа запрашивает строку у пользователя и сохраняет её в файл readme.txt. Если файл не существует, то строка не записывается.

```

timursimdyanov@ubuntu:~/work/arch-pc/lab10$
timursimdyanov@ubuntu:~/work/arch-pc/lab10$ nasm -f elf lab10-1.asm
timursimdyanov@ubuntu:~/work/arch-pc/lab10$ ld -m elf_i386 -o lab10-1 lab10-1.o
timursimdyanov@ubuntu:~/work/arch-pc/lab10$ ./lab10-1
Введите строку для записи в файл: Timur
timursimdyanov@ubuntu:~/work/arch-pc/lab10$ cat readme.txt
Timur
timursimdyanov@ubuntu:~/work/arch-pc/lab10$

```

Рис. 2.2: Запуск программы lab10-1.asm

Используя команду `chmod`, изменил права доступа к исполняемому файлу `lab10-1`, запретив его выполнение. Попытался запустить файл.

Программа не запускается, так как были убраны права на выполнение (атрибуток снят для всех пользователей).

```

timursimdyanov@ubuntu:~/work/arch-pc/lab10$
timursimdyanov@ubuntu:~/work/arch-pc/lab10$ chmod -x lab10-1
timursimdyanov@ubuntu:~/work/arch-pc/lab10$ ./lab10-1
bash: ./lab10-1: Permission denied
timursimdyanov@ubuntu:~/work/arch-pc/lab10$

```

Рис. 2.3: Запрещенный запуск файла

С помощью команды `chmod` добавил права на исполнение к файлу `lab10-1.asm`, содержащему исходный код программы. Попытался выполнить файл.

Терминал воспринимает содержимое файла как набор консольных команд. Поскольку инструкции ассемблера не являются командами оболочки, появляются ошибки. Однако, если в файл добавить команды оболочки, их можно будет выполнить.


```
timursimdyanov@Ubuntu:~/work/arch-pc/lab10$ chmod +x lab10-1.asm
timursimdyanov@Ubuntu:~/work/arch-pc/lab10$ ./lab10-1.asm
./lab10-1.asm: line 1: fg: no job control
./lab10-1.asm: line 2: SECTION: command not found
./lab10-1.asm: line 3: filename: command not found
./lab10-1.asm: line 3: Имя: command not found
./lab10-1.asm: line 4: msg: command not found
./lab10-1.asm: line 4: Сообщение: command not found
./lab10-1.asm: line 5: SECTION: command not found
./lab10-1.asm: line 6: contents: command not found
./lab10-1.asm: line 6: переменная: command not found
./lab10-1.asm: line 7: SECTION: command not found
./lab10-1.asm: line 8: global: command not found
./lab10-1.asm: line 9: _start:: command not found
./lab10-1.asm: line 10: syntax error near unexpected token `;'
./lab10-1.asm: line 10: `'; --- Печать сообщения `msg`'
timursimdyanov@Ubuntu:~/work/arch-pc/lab10$
```

Рис. 2.4: Файл lab10-1.asm с правами на выполнение

Изменил права доступа к файлам readme в соответствии с вариантом в таблице 10.4. Проверил результат с помощью команды `ls -l`.

Для варианта 4: права доступа -w- — -w- и в восьмеричном представлении 001 011 110.

```
timursimdyanov@Ubuntu:~/work/arch-pc/lab10$
timursimdyanov@Ubuntu:~/work/arch-pc/lab10$ ls -l | grep readme
-rw-rw-r-- 1 timursimdyanov timursimdyanov 0 Dec 15 02:54 readme-2.txt
-rw-rw-r-- 1 timursimdyanov timursimdyanov 6 Dec 15 02:55 readme.txt
timursimdyanov@Ubuntu:~/work/arch-pc/lab10$ chmod 202 readme.txt
timursimdyanov@Ubuntu:~/work/arch-pc/lab10$ chmod 136 readme-2.txt
timursimdyanov@Ubuntu:~/work/arch-pc/lab10$ ls -l | grep readme
---x-wxrw- 1 timursimdyanov timursimdyanov 0 Dec 15 02:54 readme-2.txt
--w-----w- 1 timursimdyanov timursimdyanov 6 Dec 15 02:55 readme.txt
timursimdyanov@Ubuntu:~/work/arch-pc/lab10$
```

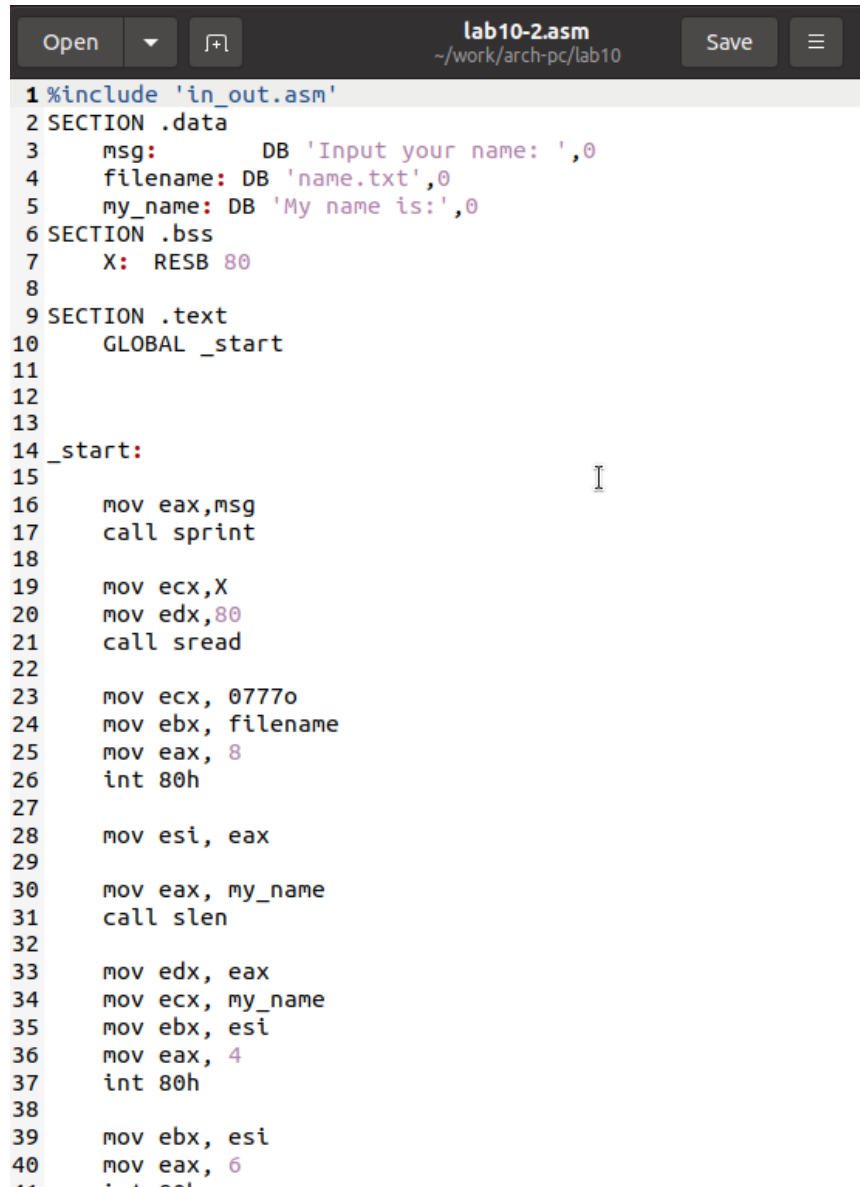
Рис. 2.5: Настройка прав доступа к файлам

2.1 Задание для самостоятельной работы

Написал программу, которая реализует следующий алгоритм:

1. Выводит приглашение: “Как Вас зовут?”.
2. Вводит с клавиатуры фамилию и имя.
3. Создает файл name.txt.

4. Записывает в файл сообщение “Меня зовут”.
5. Дописывает в файл строку, введенную пользователем.
6. Закрывает файл.



```
1 %include 'in_out.asm'
2 SECTION .data
3     msg:      DB 'Input your name: ',0
4     filename: DB 'name.txt',0
5     my_name:  DB 'My name is:',0
6 SECTION .bss
7     X:  RESB 80
8
9 SECTION .text
10     GLOBAL _start
11
12
13
14 _start:
15
16     mov eax,msg
17     call sprint
18
19     mov ecx,X
20     mov edx,80
21     call sread
22
23     mov ecx, 0777o
24     mov ebx, filename
25     mov eax, 8
26     int 80h
27
28     mov esi, eax
29
30     mov eax, my_name
31     call slen
32
33     mov edx, eax
34     mov ecx, my_name
35     mov ebx, esi
36     mov eax, 4
37     int 80h
38
39     mov ebx, esi
40     mov eax, 6
41     int 80h
```

Рис. 2.6: Исходный код программы lab10-2.asm

```
timursimdyanov@Ubuntu:~/work/arch-pc/lab10$  
timursimdyanov@Ubuntu:~/work/arch-pc/lab10$ nasm -f elf lab10-2.asm  
timursimdyanov@Ubuntu:~/work/arch-pc/lab10$ ld -m elf_i386 -o lab10-2 lab10-2.o  
timursimdyanov@Ubuntu:~/work/arch-pc/lab10$ ./lab10-2  
Input your name: Timur  
timursimdyanov@Ubuntu:~/work/arch-pc/lab10$ cat name.txt  
My name is:Timur  
timursimdyanov@Ubuntu:~/work/arch-pc/lab10$
```

Рис. 2.7: Результат выполнения программы lab10-2.asm

3 Выводы

Изучены основные операции работы с файлами и управление правами доступа.