

Отчёт по лабораторной работе №10

Дисциплина: архитектура компьютера

Ахмад мд Шешир

Содержание

1	Цель работы	5
2	Лабораторная работа	6
3	Самостоятельная работа	9
3.1	Программа отработала корректно!!	11
3.2	Текст программы в самостоятельной работе	11
4	Выводы	14

Список иллюстраций

2.1	Создание директории	6
2.2	Редактирование текста	7
2.3	Запуск исполняемого файла	7
2.4	Доступ к файлу	8
2.5	Предоставляю права доступа к файлу	8
2.6	Предоставляю права доступа к файлу	8
3.1	Создание файла для самостоятельной работы	9
3.2	Написание программы для самостоятельной работы	10
3.3	Исполнение файла	11

Список таблиц

1 Цель работы

Приобретение навыков написания программ для работы с файлами.

2 Лабораторная работа

Шаг 1

С помощью утилиты `mkdir` создаю директорию `lab10`, перехожу в нее и создаю файл для работы.

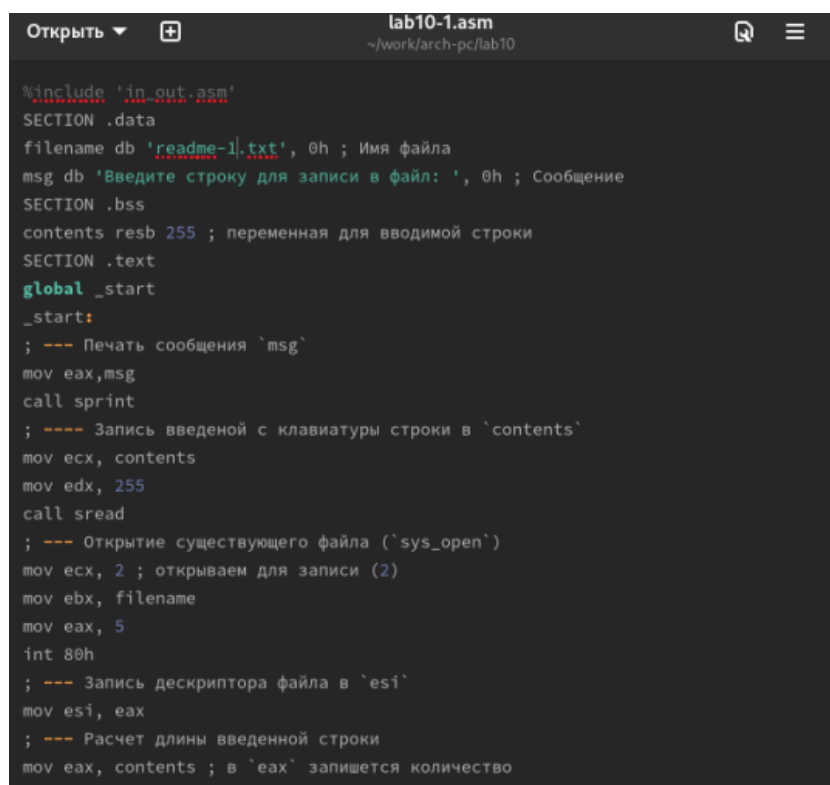
A screenshot of a terminal window with a dark background. The title bar at the top shows a plus icon, the text "sheshir.m@fedora:~/work/arch-pc/lab10", and a search icon. The terminal content shows the following commands and their outputs:

```
[sheshir.m@fedora ~]$ mkdir ~/work/arch-pc/lab10
[sheshir.m@fedora ~]$ cd ~/work/arch-pc/lab10
[sheshir.m@fedora lab10]$ touch lab10-1.asm readme-1.txt readme-2.txt
[sheshir.m@fedora lab10]$
```

Рис. 2.1: Создание директории

Шаг 2

Открываю созданный файл `lab10-1.asm`, вставляю в него программу из Листин-
га



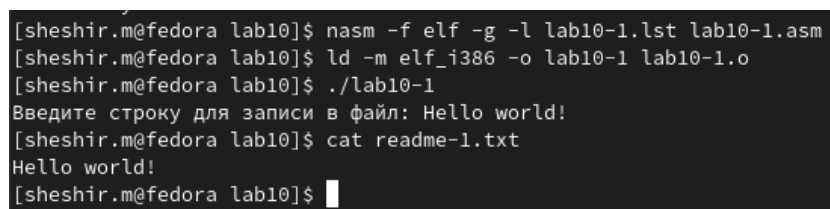
```
lab10-1.asm
~/work/arch-pc/lab10

%include 'in_out.asm'
SECTION .data
filename db 'readme-1.txt', 0h ; Имя файла
msg db 'Введите строку для записи в файл: ', 0h ; Сообщение
SECTION .bss
contents resb 255 ; переменная для вводимой строки
SECTION .text
global _start
_start:
; --- Печать сообщения `msg`
mov eax, msg
call sprint
; ---- Запись введенной с клавиатуры строки в `contents`
mov ecx, contents
mov edx, 255
call sread
; --- Открытие существующего файла (`sys_open`)
mov ecx, 2 ; открываем для записи (2)
mov ebx, filename
mov eax, 5
int 80h
; --- Запись дескриптора файла в `esi`
mov esi, eax
; --- Расчет длины введенной строки
mov eax, contents ; в `eax` запишется количество
```

Рис. 2.2: Редактирование текста

Шаг 3

Создаю исполняемый файл программы, а также сам файл, куда будет записываться строка.



```
[sheshir.m@fedora lab10]$ nasm -f elf -g -l lab10-1.lst lab10-1.asm
[sheshir.m@fedora lab10]$ ld -m elf_i386 -o lab10-1 lab10-1.o
[sheshir.m@fedora lab10]$ ./lab10-1
Введите строку для записи в файл: Hello world!
[sheshir.m@fedora lab10]$ cat readme-1.txt
Hello world!
[sheshir.m@fedora lab10]$
```

Рис. 2.3: Запуск исполняемого файла

Шаг 4

С помощью команды `chmod` изменяю права доступа к исполняемому файлу `lab10-1`, запретив его выполнение.

```
[sheshir.m@fedora lab10]$ chmod 137 lab10-1
[sheshir.m@fedora lab10]$ cat lab10-1
cat: lab10-1: Отказано в доступе
[sheshir.m@fedora lab10]$
```

Рис. 2.4: Доступ к файлу

Шаг 5

В соответствии со своим 20-ым вариантом в таблице, предоставляю права доступа к файлу `readme-1.txt` представленные в символьном виде: `— rw- -w-`

```
[sheshir.m@fedora lab10]$ chmod go+w readme-1.txt
[sheshir.m@fedora lab10]$ ls -l readme-1.txt
-rw-rw-rw-. 1 sheshir.m dayanchberdyev 13 окт 30 05:09 readme-1.txt
[sheshir.m@fedora lab10]$ chmod 137 readme-2.txt
[sheshir.m@fedora lab10]$ ls -l readme-2.txt
---x-wxrw-. 1 sheshir.m dayanchberdyev 0 окт 30 05:05 readme-2.txt
[sheshir.m@fedora lab10]$
```

Рис. 2.5: Предоставляю права доступа к файлу

Шаг 6

Предоставляю права доступа к файлу `readme-2.txt` представленные в символьном – в двоичном виде: `001 011 111`, то есть `1 3 7`

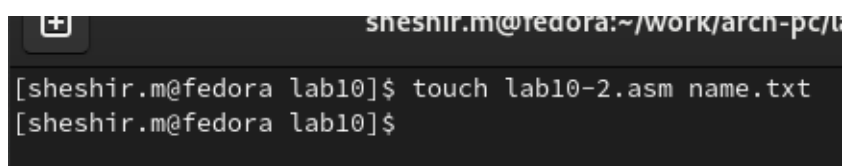
```
[sheshir.m@fedora lab10]$ chmod go+w readme-1.txt
[sheshir.m@fedora lab10]$ ls -l readme-1.txt
-rw-rw-rw-. 1 sheshir.m dayanchberdyev 13 окт 30 05:09 readme-1.txt
[sheshir.m@fedora lab10]$ chmod 137 readme-2.txt
[sheshir.m@fedora lab10]$ ls -l readme-2.txt
---x-wxrw-. 1 sheshir.m dayanchberdyev 0 окт 30 05:05 readme-2.txt
[sheshir.m@fedora lab10]$
```

Рис. 2.6: Предоставляю права доступа к файлу

3 Самостоятельная работа

Шаг 1

Создаю новый файлы для самостоятельной работы

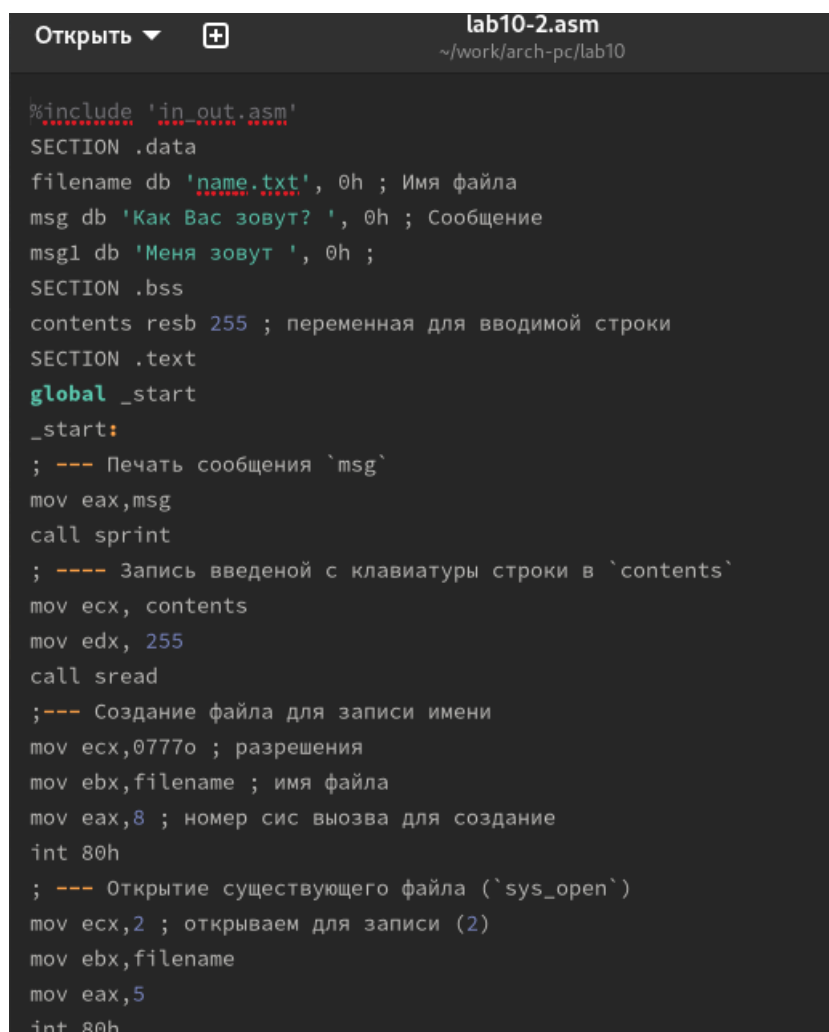
A terminal window with a dark background. The title bar shows a window icon and the text 'sheshir.m@fedora:~/work/arch-pc/...'. The terminal content shows two lines: the first line is '[sheshir.m@fedora lab10]\$ touch lab10-2.asm name.txt' and the second line is '[sheshir.m@fedora lab10]\$'.

```
sheshir.m@fedora:~/work/arch-pc/...  
[sheshir.m@fedora lab10]$ touch lab10-2.asm name.txt  
[sheshir.m@fedora lab10]$
```

Рис. 3.1: Создание файла для самостоятельной работы

Шаг 2

Пишу в неё текст программы, которая будет запрашивать имя пользователя, будет создавать файл и записывать в нее введенное имя, добавив в начале текст : “Меня зовут”.



```
Открыть ▾ + lab10-2.asm
~/work/arch-pc/lab10

%include 'in_out.asm'
SECTION .data
filename db 'name.txt', 0h ; Имя файла
msg db 'Как Вас зовут? ', 0h ; Сообщение
msg1 db 'Меня зовут ', 0h ;
SECTION .bss
contents resb 255 ; переменная для вводимой строки
SECTION .text
global _start
_start:
; --- Печать сообщения `msg`
mov eax,msg
call sprint
; ---- Запись введенной с клавиатуры строки в `contents`
mov ecx, contents
mov edx, 255
call sread
;--- Создание файла для записи имени
mov ecx,0777o ; разрешения
mov ebx,filename ; имя файла
mov eax,8 ; номер сис вызова для создание
int 80h
; --- Открытие существующего файла (`sys_open`)
mov ecx,2 ; открываем для записи (2)
mov ebx,filename
mov eax,5
int 80h
```

Рис. 3.2: Написание программы для самостоятельной работы

Шаг 3

Создаю исполняемый файл и запускаю её, ввожу свое имя и фамилию. С помощью атрибуты cat проверяю содержимое созданного файла.

```
sheshir.m@fedora:~/work/arch-pc/lab10
[sheshir.m@fedora lab10]$ touch lab10-2.asm name.txt
[sheshir.m@fedora lab10]$ ls
in_out.asm  lab10-1.asm  lab10-1.o  name.txt  readme-2.txt
lab10-1     lab10-1.lst  lab10-2.asm  readme-1.txt
[sheshir.m@fedora lab10]$ nasm -f elf -g -l lab10-2.lst lab10-2.asm
[sheshir.m@fedora lab10]$ ld -m elf_i386 -o lab10-2 lab10-2.o
[sheshir.m@fedora lab10]$ ./lab10-2
Как Вас зовут? Ахмад мд Шешир
[sheshir.m@fedora lab10]$ ls -l
итого 84
-rw-r--r--. 1 sheshir.m dayanchberdyev 3942 ноя 12 2023 in_out.asm
---x-wxrwX. 1 sheshir.m dayanchberdyev 9744 окт 30 05:08 lab10-1
-rw-r--r--. 1 sheshir.m dayanchberdyev 1142 дек 15 2023 lab10-1.asm
-rw-r--r--. 1 sheshir.m dayanchberdyev 13448 окт 30 05:08 lab10-1.lst
-rw-r--r--. 1 sheshir.m dayanchberdyev 2528 окт 30 05:08 lab10-1.o
-rwxr-xr-x. 1 sheshir.m dayanchberdyev 9760 окт 30 05:16 lab10-2
-rw-r--r--. 1 sheshir.m dayanchberdyev 1559 дек 16 2023 lab10-2.asm
-rw-r--r--. 1 sheshir.m dayanchberdyev 14397 окт 30 05:15 lab10-2.lst
-rw-r--r--. 1 sheshir.m dayanchberdyev 2608 окт 30 05:15 lab10-2.o
-rw-r--r--. 1 sheshir.m dayanchberdyev 47 окт 30 05:16 name.txt
-rw-rw-rw-. 1 sheshir.m dayanchberdyev 13 окт 30 05:09 readme-1.txt
---x-wxrwX. 1 sheshir.m dayanchberdyev 0 окт 30 05:05 readme-2.txt
[sheshir.m@fedora lab10]$ cat name.txt
Меня зовут Ахмад мд Шешир
[sheshir.m@fedora lab10]$
```

Рис. 3.3: Исполнение файла

3.1 Программа отработала корректно!!

3.2 Текст программы в самостоятельной работе

```
%include 'in_out.asm'

SECTION .data

filename db 'name.txt', 0h ; Имя файла
msg db 'Как Вас зовут? ', 0h ; Сообщение
msg1 db 'Меня зовут ', 0h ;

SECTION .bss

contents resb 255 ; переменная для вводимой строки

SECTION .text

global _start

_start:
```

```

; --- Печать сообщения `msg`
mov eax,msg
call sprint
; ---- Запись введенной с клавиатуры строки в `contents`
mov ecx, contents
mov edx, 255
call sread
;--- Создание файла для записи имени
mov ecx,0777o ; разрешения
mov ebx,filename ; имя файла
mov eax,8 ; номер сис вызова для создание
int 80h
; --- Открытие существующего файла (`sys_open`)
mov ecx,2 ; открываем для записи (2)
mov ebx,filename
mov eax,5
int 80h
; --- Запись дескриптора файла в `esi`
mov esi,eax
; --- Расчет длины введенной строки msg1
mov eax,msg1
call slen
; --- Записываем в файл `msg1` (`sys_write`)
mov edx, eax
mov ecx, msg1
mov ebx, esi
mov eax, 4
int 80h
; --- Расчет длины введенной строки

```

```
mov eax, contents ; в `eax` запишется количество
call slen ; введенных байтов
; --- Записываем в файл `contents` (`sys_write`)
mov edx, eax
mov ecx, contents
mov ebx, esi
mov eax, 4
int 80h
; --- Закрываем файл (`sys_close`)
mov ebx, esi
mov eax, 6
int 80h
call quit
```

4 Выводы

В ходе выполнения работы были получены навыки по работе с файлами в NASM.