

Отчёт по лабораторной работе №10

Дисциплина: архитектура компьютера

Чернятьева Олеся Олеговна

Содержание

1	Цель работы	4
2	Выполнение лабораторной работы	5
3	Самостоятельная работа	10
4	Вывод	15

Список иллюстраций

2.1	Создание директории	5
2.2	Редактирование текста	6
2.3	Запуск исполняемого файла	7
2.4	Доступ к файлу	7
2.5	Изменение прав к файлу	8
2.6	Предоставляю права доступа к файлу	8
2.7	Предоставляю права доступа к файлу	9
3.1	Создание файла для самостоятельной	10
3.2	Написание программы для самостоятельной	11
3.3	Исполнение файла	11
3.4	вставляю программу	12


1 Цель работы

Приобретение навыков написания программ для работы с файлами в NASM.

2 Выполнение лабораторной работы

Шаг 1

С помощью утилиты `mkdir` создаю директорию `lab10`, перехожу в нее и создаю файл для работы. (рис. [2.1])

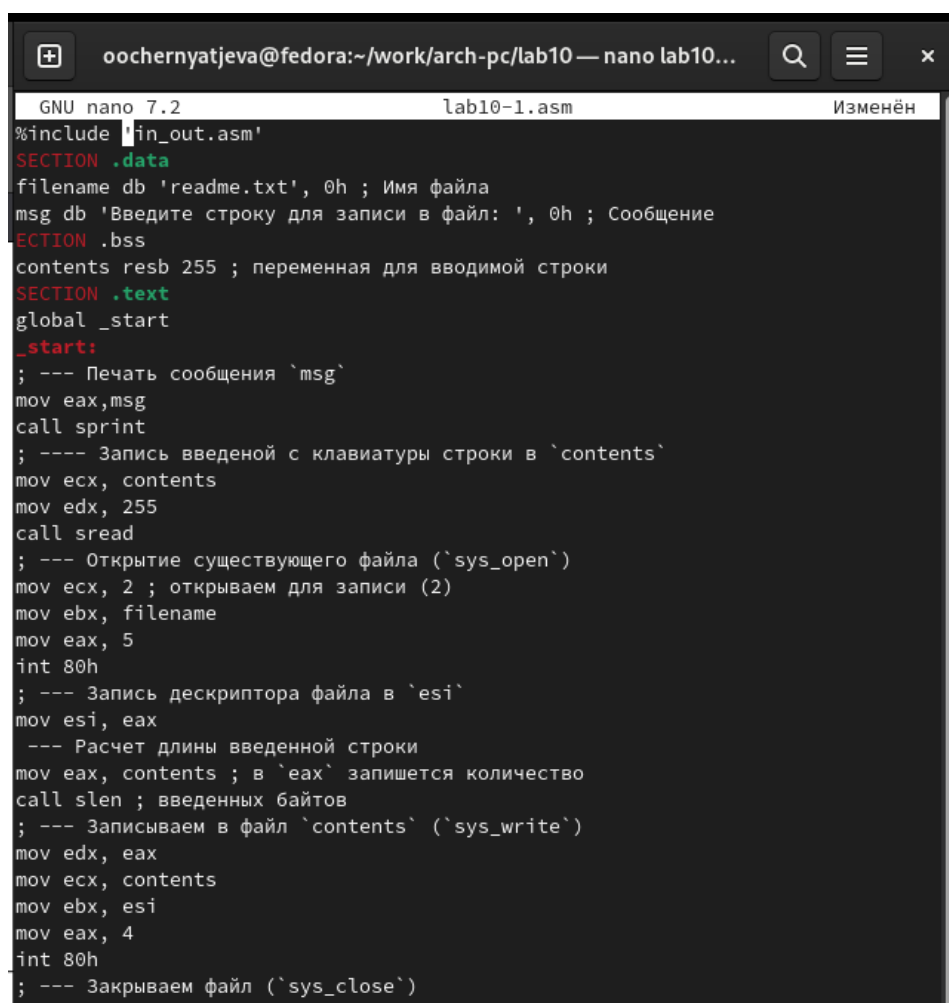


```
oochernyatjeva@fedora:~/work/arch-pc/lab10
[oochernyatjeva@fedora ~]$ mkdir ~/work/arch-pc/lab10
[oochernyatjeva@fedora ~]$ cd ~/work/arch-pc/lab10
[oochernyatjeva@fedora lab10]$ touch lab10-1.asm readme-1.txt readme-2.txt
[oochernyatjeva@fedora lab10]$
```

Рис. 2.1: Создание директории

Шаг 2

Открываю созданный файл `lab10-1.asm`, вставляю в него программу из Листинга.(рис. [2.2]).



```
GNU nano 7.2 lab10-1.asm Изменён
%include 'in_out.asm'
SECTION .data
filename db 'readme.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
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`)
```

Рис. 2.2: Редактирование текста

Шаг 3

Создаю исполняемый файл программы и ввожу свое имя и фамилию, и сразу проверяю работу (рис. [2.3]).

```
[oochernyatjeva@fedora lab10]$ ld -m elf_i386 -o lab10-1 lab10-1.o
[oochernyatjeva@fedora lab10]$ ./lab10-1
Введите строку для записи в файл: Олеся Чернятьева
[oochernyatjeva@fedora lab10]$ cat readme.txt
Олеся Чернятьева
[oochernyatjeva@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. (рис. [2.4]).

```
[oochernyatjeva@fedora lab10]$ chmod 640 lab10-1
[oochernyatjeva@fedora lab10]$ ls -l lab10-1
-rw-r-----. 1 oochernyatjeva darfonos 9164 дек 16 19:57 lab10-1
[oochernyatjeva@fedora lab10]$ ./lab10-1
bash: ./lab10-1: Отказано в доступе
[oochernyatjeva@fedora lab10]$
```

Рис. 2.4: Доступ к файлу

1. В символьном представлении есть возможность явно указывать какой группе какие права необходимо добавить или отнять. Так '6' дает право владельцу лишь для чтения и записи, но не для исполнения.
 2. '4' дает право группе лишь для чтения.
 3. '0' отнимает все права для всех прочих пользователей.
-

Шаг 5

С помощью команды `chmod` изменяю права доступа к файлу `lab10-1.asm` с исходным текстом программы, добавив права на исполнение. (рис. [2.5]).

```

[oochernyatjeva@fedora lab10]$ chmod gu+x lab10-1.asm
[oochernyatjeva@fedora lab10]$ ls -l lab10-1.asm
-rwxr-xr-x. 1 oochernyatjeva darfonos 1143 дек 16 19:56 lab10-1.asm
[oochernyatjeva@fedora lab10]$ ./lab10-1.asm
./lab10-1.asm: строка 1: fg: нет управления заданиями
./lab10-1.asm: строка 2: SECTION: команда не найдена
./lab10-1.asm: строка 3: filename: команда не найдена
./lab10-1.asm: строка 3: Имя: команда не найдена
./lab10-1.asm: строка 4: msg: команда не найдена
./lab10-1.asm: строка 4: Сообщение: команда не найдена
./lab10-1.asm: строка 5: SECTION: команда не найдена
./lab10-1.asm: строка 6: contents: команда не найдена
./lab10-1.asm: строка 6: переменная: команда не найдена
./lab10-1.asm: строка 7: SECTION: команда не найдена
./lab10-1.asm: строка 8: global: команда не найдена
./lab10-1.asm: строка 9: _start:: команда не найдена
./lab10-1.asm: строка 10: синтаксическая ошибка рядом с неожиданным маркером «
;»
./lab10-1.asm: строка 10: `; --- Печать сообщения `msg` '
[oochernyatjeva@fedora lab10]$

```

Рис. 2.5: Изменение прав к файлу

- С Помощью аргументов “u” и “+” добавляем право владельцу на исполнение файла, однако в файле находится исходный код программы, который еще не ассемблирован, и его выполнение невозможно.
- Происходит исполнение файла, но она не будет работать.

Шаг 6

В соответствии со своим 10-ым вариантом в таблице, предоставляю права доступа к файлу readme-1.txt представленные в символьном виде. (рис. [2.6]).

```

[oochernyatjeva@fedora lab10]$ chmod 447 readme-1.txt
[oochernyatjeva@fedora lab10]$ ls -l readme-1.txt
-r--r--rwx. 1 oochernyatjeva darfonos 0 дек 16 19:48 readme-1.txt
[oochernyatjeva@fedora lab10]$

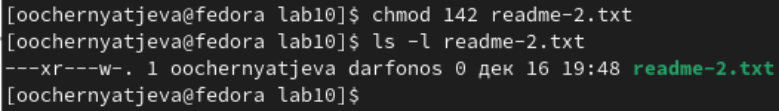
```

Рис. 2.6: Предоставляю права доступа к файлу

- Вначале проверяю изначальные права, которые имеют пользователи к файлу по умолчанию, затем даю ей другие права в соответствии с вариатном.

Шаг 7

Предоставляю права доступа к файлу `readme-2.txt` представленные в символьном – в двоичном виде. (рис. [2.7]).



```
[oochernyatjeva@fedora lab10]$ chmod 142 readme-2.txt
[oochernyatjeva@fedora lab10]$ ls -l readme-2.txt
---xr---w-. 1 oochernyatjeva darfonos 0 дек 16 19:48 readme-2.txt
[oochernyatjeva@fedora lab10]$
```

Рис. 2.7: Предоставляю права доступа к файлу

3 Самостоятельная работа

Шаг 1

Создаю новый файл для самостоятельной работы(рис.[3.1]).

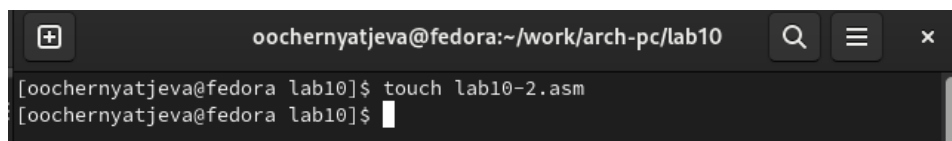
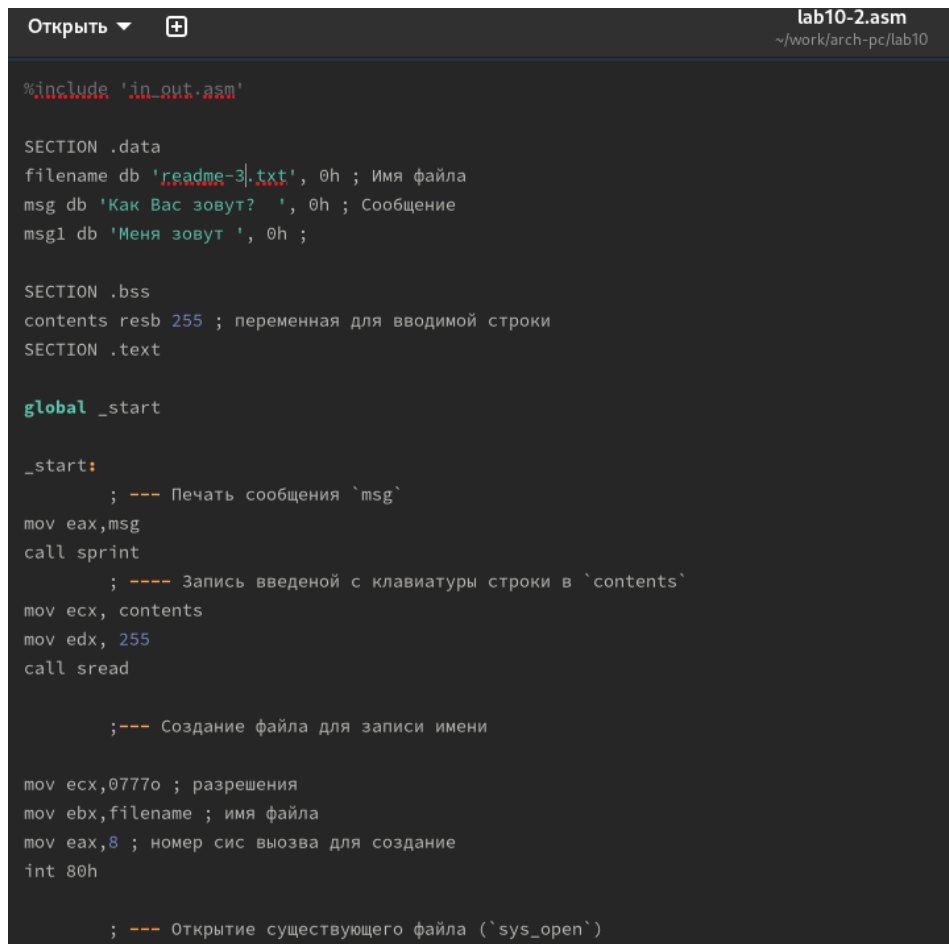
A screenshot of a terminal window. The title bar shows the user 'oochernyatjeva@fedora' and the current directory '~/work/arch-pc/lab10'. The terminal contains two lines of text: '[oochernyatjeva@fedora lab10]\$ touch lab10-2.asm' and a second line '[oochernyatjeva@fedora lab10]\$' with a cursor at the end. The window has standard Linux window controls (minimize, maximize, close) and a search icon.

Рис. 3.1: Создание файла для самостоятельной

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”. (рис. [3.2]).



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

%include 'in_out.asm'

SECTION .data
filename db 'readme-3.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

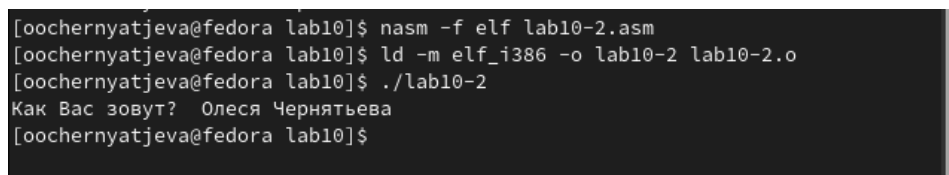
    mov ecx,0777o ; разрешения
    mov ebx,filename ; имя файла
    mov eax,8 ; номер сис вызова для создание
    int 80h

    ; --- Открытие существующего файла (`sys_open`)
```

Рис. 3.2: Написание программы для самостоятельной

Шаг 3

Создаю исполняемый файл и запускаю её, ввожу свое имя и фамилию. (рис. [3.3]).

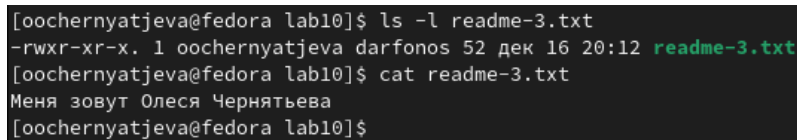


```
[oochernyatjeva@fedora lab10]$ nasm -f elf lab10-2.asm
[oochernyatjeva@fedora lab10]$ ld -m elf_i386 -o lab10-2 lab10-2.o
[oochernyatjeva@fedora lab10]$ ./lab10-2
Как Вас зовут? Олеся Чернятьева
[oochernyatjeva@fedora lab10]$
```

Рис. 3.3: Исполнение файла

Шаг 4

Сразу проверяю с помощью атрибуты **ls** созданся ли необходимый файл, куда должны записываться введенные символы. С помощью атрибуты **cat** проверяю содержимое созданного файла. (рис. [3.4]).



```
[oochernyatjeva@fedora lab10]$ ls -l readme-3.txt
-rwxr-xr-x. 1 oochernyatjeva darfonos 52 дек 16 20:12 readme-3.txt
[oochernyatjeva@fedora lab10]$ cat readme-3.txt
Меня зовут Олеся Чернятьева
[oochernyatjeva@fedora lab10]$
```

Рис. 3.4: вставляю программу

Программа отработала корректно!!

Текст программы в самостоятельной работе

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

```
SECTION .data
```

```
filename db 'readme-3.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 Вывод

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