

Лабораторная работа №10

Работа с файлами средствами Nasm

Павличенко Родион

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
3	Выполнение задания для самостоятельной работы	10
4	Выводы	13

Список иллюстраций

2.1	Создание рабочей директории и файлов	6
2.2	Запуск Midnight commander	6
2.3	Вставка кода из файла листинга 9.1	7
2.4	Копирование файла in_out.asm в рабочую директорию	7
2.5	Сборка программы из файла lab10-1.asm и её запуск	8
2.6	Повторный запуск файла	8
2.7	Запуск файла lab10-1.asm	8
2.8	Изменение прав доступа и проверка	9
3.1	Создание файла lab10-2.asm	10
3.2	Запуск Midnight commander	10
3.3	Код файла самостоятельной работы	11
3.4	Запуск файла и проверка корректности работы	12

Список таблиц

1 Цель работы

Приобретение навыков написания программ для работы с файлами.

2 Выполнение лабораторной работы

Для начала выполнения работы необходимо создать рабочую папку и файл lab10-1.asm, а также файлы readme-1.txt и readme-2.txt :

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

Рис. 2.1: Создание рабочей директории и файлов

Далее, запустим Midnight commander:

```
rapavlichenko@rapavlichenko:~/work/arch-pc/lab10$ mc
```

Рис. 2.2: Запуск Midnight commander

Вставим в файл lab10-1.asm код из листинга 10.1 :

```
GNU nano 7.2 /home/rapavlichenko/work/arch-pc/lab10/lab10-1.asm Изменён
;-----
; Запись в файл строки введенной на запрос
;-----
#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`)
```

Рис. 2.3: Вставка кода из файла листинга 9.1

Скопируем файл in_out.asm из директории прошлой работы :

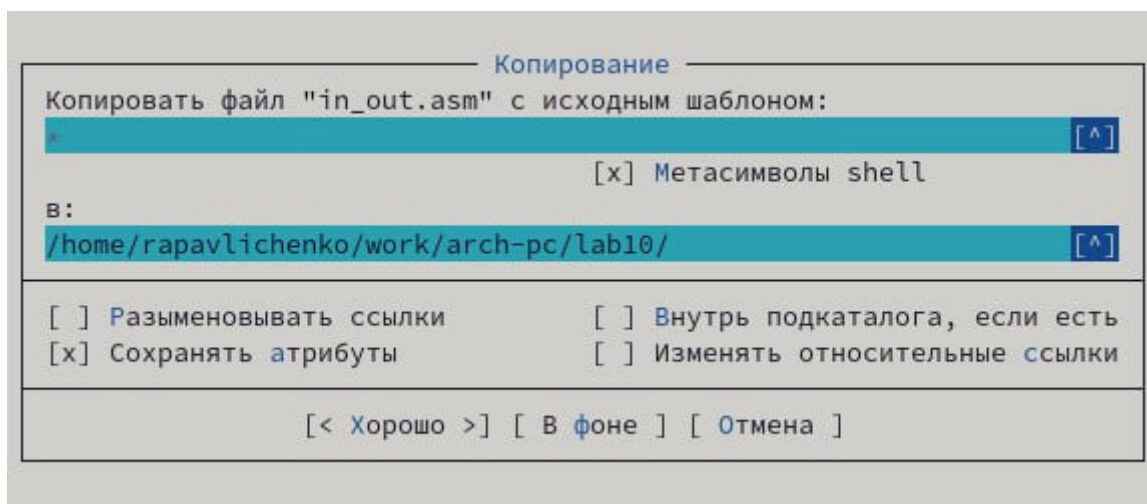


Рис. 2.4: Копирование файла in_out.asm в рабочую директорию

Соберём программу и посмотрим на результат :

```

rapavlichenko@rapavlichenko:~/work/arch-pc/lab10$ nasm -f elf lab10-1.asm
rapavlichenko@rapavlichenko:~/work/arch-pc/lab10$ ld -m elf_i386 -o lab10-1 lab10-1.o
rapavlichenko@rapavlichenko:~/work/arch-pc/lab10$ ./lab10-1
Введите строку для записи в файл: hello world
rapavlichenko@rapavlichenko:~/work/arch-pc/lab10$

```

Рис. 2.5: Сборка программы из файла lab10-1.asm и её запуск

Как видим, файл выполнен, однако, ничего не произошло, так как в коде прописано записать данные в файл readme.txt, которого не существует. Теперь попробуем изменить права доступа для программы lab10-1 так, чтобы запретить всем группам пользователей запускать данную программу и попробуем запустить его :

```

rapavlichenko@rapavlichenko:~/work/arch-pc/lab10$ chmod ugo-x lab10-1
rapavlichenko@rapavlichenko:~/work/arch-pc/lab10$ ./lab10-1
bash: ./lab10-1: Отказано в доступе
rapavlichenko@rapavlichenko:~/work/arch-pc/lab10$

```

Рис. 2.6: Повторный запуск файла

Как видим, файл даже не запустился, и нам вывело ошибку “Отказано в доступе”. Теперь попробуем добавить файлу с исходным кодом lab10-1.asm права на запуск, и попробуем его запустить :

```

rapavlichenko@rapavlichenko:~/work/arch-pc/lab10$ chmod ugo+x lab10-1.asm
rapavlichenko@rapavlichenko:~/work/arch-pc/lab10$ ./lab10-1.asm
./lab10-1.asm: строка 1: синтаксическая ошибка рядом с неожиданным маркером «;»
./lab10-1.asm: строка 1: `;-----'
rapavlichenko@rapavlichenko:~/work/arch-pc/lab10$

```

Рис. 2.7: Запуск файла lab10-1.asm

Как видим, нам вывело много ошибок, так как файл сам по себе не предназначен для запуска. ведь это файл с исходным кодом, который требует предварительной сборки. Теперь попробуем выдать права доступа для файлов readme-1.txt и readme-2.txt согласно варианту 1. Так, мы используем chmod и пишем права доступа в восьмиричном виде :


```

rapavlichenko@rapavlichenko:~/work/arch-pc/lab10$ chmod 137 readme-1.txt
rapavlichenko@rapavlichenko:~/work/arch-pc/lab10$ chmod 862 readme-2.txt
chmod: неверный режим: «862»
По команде «chmod --help» можно получить дополнительную информацию.
rapavlichenko@rapavlichenko:~/work/arch-pc/lab10$ chmod 062 readme-2.txt
rapavlichenko@rapavlichenko:~/work/arch-pc/lab10$ ls -l
итого 24
-rw-r--r--. 1 rapavlichenko rapavlichenko 3942 окт 19 12:39 in_out.asm
-rw-r--r--. 1 rapavlichenko rapavlichenko 9164 ноя 24 18:04 lab10-1
-rwxr-xr-x. 1 rapavlichenko rapavlichenko 1287 ноя 24 18:03 lab10-1.asm
-rw-r--r--. 1 rapavlichenko rapavlichenko 1472 ноя 24 18:04 lab10-1.o
---x-wrwx. 1 rapavlichenko rapavlichenko    0 ноя 24 18:00 readme-1.txt
----rw--w-. 1 rapavlichenko rapavlichenko    0 ноя 24 18:00 readme-2.txt
rapavlichenko@rapavlichenko:~/work/arch-pc/lab10$

```

Рис. 2.8: Изменение прав доступа и проверка

Как видим, права доступа установились корректно.

3 Выполнение задания для самостоятельной работы

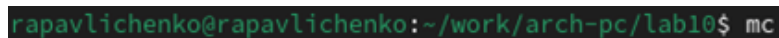
Далее, напишем код согласно заданию к самостоятельной работе. Он должен создать файл name.txt, записать туда фразу “Меня зовут”, запросить фамилию и имя пользователя и дописать их в файл, но сначала создадим файл



```
rapavlichenko@rapavlichenko:~/work/arch-pc/lab10$ touch lab10-2.asm
```

Рис. 3.1: Создание файла lab10-2.asm

Далее, запустим Midnight commander



```
rapavlichenko@rapavlichenko:~/work/arch-pc/lab10$ mc
```

Рис. 3.2: Запуск Midnight commander

Напишем код

```

GNU nano 7.2 /home/rapavlichenko/work/arch-pc/lab10/lab10-2.asm Изменён
#include 'in_out.asm'
SECTION .data
filename db 'name.txt', 0h ; Имя файла
msg db 'Как вас зовут? ', 0h ; Сообщение
myname 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
mov ecx, 0777o
mov ebx, filename
mov eax, 8
int 80h
mov edx, 255 ; количество байтов для записи
mov ecx, myname ; адрес строки для записи в файл
mov ebx, eax ; дескриптор файла
mov eax, 4 ; номер системного вызова sys_write
int 80h ; вызов ядра
mov eax, 5
mov edx, 2 ; значение смещения -- конец файла
mov ecx, 0 ; смещение на 0 байт
mov ebx, eax ; дескриптор файла
mov eax, 19 ; номер системного вызова sys_lseek
int 80h ; вызов ядра
mov edx, 9 ; Запись в конец файла
mov ecx, contents ; строки из переменной msg
mov eax, 4
int 80h
; --- Запись дескриптора файла в esi
; --- Расчет длины введенной строки
; --- Закрываем файл (sys_close)
mov ebx, esi
mov eax, 6
int 80h
call quit

```

Рис. 3.3: Код файла самостоятельной работы

Теперь соберём программу и проверим корректность выполнения её рабо-

ты. Для этого с помощью `ls` мы проверим, созданся ли файл, а с помощью `cat` посмотрим, что в файл записалось то, что нужно :

```
rapavlichenko@rapavlichenko:~/work/arch-pc/lab10$ nasm -f elf lab10-2.asm
rapavlichenko@rapavlichenko:~/work/arch-pc/lab10$ ld -m elf_i386 -o lab10-2 lab10-2.o
rapavlichenko@rapavlichenko:~/work/arch-pc/lab10$ ./lab10-2
Как вас зовут? Павличенко Родион
rapavlichenko@rapavlichenko:~/work/arch-pc/lab10$ ls
in_out.asm  lab10-1.asm  lab10-2      lab10-2.o  readme-1.txt
lab10-1     lab10-1.o   lab10-2.asm  name.txt   readme-2.txt
rapavlichenko@rapavlichenko:~/work/arch-pc/lab10$ cat name.txt
Меня зовут Павличенко Родион
rapavlichenko@rapavlichenko:~/work/arch-pc/lab10$
```

Рис. 3.4: Запуск файла и проверка корректности работы

Как видим, программа выполнилась корректно

4 Выводы

В результате выполнения лабораторной работы я приобрел навыки написания программ для работы с файлами.