

Лабораторная работа №10

Работа с файлами средствами Nasm

Богданюк Анна Васильевна

Содержание

1	Цель работы	5
2	Задание	6
3	Теоретическое введение	7
4	Выполнение лабораторной работы	9
5	Выводы	17
	Список литературы	18

Список таблиц

Список иллюстраций

4.1	Создаю каталог и файлы	9
4.2	Файл lab10-1.asm	10
4.3	Создание, запуск и проверка	12
4.4	Изменение права доступа, проверка	12
4.5	Изменение права доступа lab10-1.asm	12
4.6	Запуск lab10-1.asm	13
4.7	Изменение права доступа в символьном виде	13
4.8	Изменение права доступа в двоичном виде	13
4.9	Файл lab10-2.asm	14
4.10	Создание и запуск	16
4.11	Проверка	16

1 Цель работы

Приобретение навыков написания программ для работы с файлами.

2 Задание

1. Выполнение лабораторной работы
2. Задания для самостоятельной работы

3 Теоретическое введение

ОС GNU/Linux является многопользовательской операционной системой. И для обеспечения защиты данных одного пользователя от действий других пользователей существуют специальные механизмы разграничения доступа к файлам. Кроме ограничения доступа, данный механизм позволяет разрешить другим пользователям доступ данным для совместной работы. Права доступа определяют набор действий (чтение, запись, выполнение), разрешённых для выполнения пользователям системы над файлами. Для каждого файла пользователь может входить в одну из трех групп: владелец, член группы владельца, все остальные. Для каждой из этих групп может быть установлен свой набор прав доступа. Владелец файла является его создатель. Для предоставления прав доступа другому пользователю или другой группе командой `chown [ключи] [:новая_группа]` или `chgrp [ключи] < новая_группа >`

Общий алгоритм работы с системными вызовами в Nasm можно представить в следующем виде: 1. Поместить номер системного вызова в регистр EAX; 2. Поместить аргументы системного вызова в регистрах EBX, ECX и EDX; 3. Вызов прерывания (int 80h); 4. Результат обычно возвращается в регистр EAX.

Для создания и открытия файла служит системный вызов `sys_creat`, который использует следующие аргументы: права доступа к файлу в регистре ECX, имя файла в EBX и номер системного вызова `sys_creat` (8) в EAX

Для записи в файл служит системный вызов `sys_write`, который использует следующие аргументы: количество байтов для записи в регистре EDX, строку содержимого для записи ECX, файловый дескриптор в EBX и номер системного

вызова `sys_write` (4) в `EAX`. Системный вызов возвращает фактическое количество записанных байтов в регистр `EAX`. В случае ошибки, код ошибки также будет находиться в регистре `EAX`. Прежде чем записывать в файл, его необходимо создать или открыть, что позволит получить дескриптор файла.

Для чтения данных из файла служит системный вызов `sys_read`, который использует следующие аргументы: количество байтов для чтения в регистре `EDX`, адрес в памяти для записи прочитанных данных в `ECX`, файловый дескриптор в `EBX` и номер системного вызова `sys_read` (3) в `EAX`. Как и для записи, прежде чем читать из файла, его необходимо открыть, что позволит получить дескриптор файла.

Для правильного закрытия файла служит системный вызов `sys_close`, который использует один аргумент – дескриптор файла в регистре `EBX`. После вызова ядра происходит удаление дескриптора файла, а в случае ошибки, системный вызов возвращает код ошибки в регистр `EAX`.

Для изменения содержимого файла служит системный вызов `sys_lseek`, который использует следующие аргументы: исходная позиция для смещения `EDX`, значение смещения в байтах в `ECX`, файловый дескриптор в `EBX` и номер системного вызова `sys_lseek` (19) в `EAX`.

Удаление файла осуществляется системным вызовом `sys_unlink`, который использует один аргумент – имя файла в регистре `EBX`.

4 Выполнение лабораторной работы

1. Выполнение лабораторной работы

Создаю каталог lab10 для программ лабораторной работы №10, перехожу в него и создаю файлы lab10-1.asm, readme-1.txt и readme-2.txt (рис. 4.1).

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

Рис. 4.1: Создаю каталог и файлы

Ввожу в файл lab10-1.asm текст программы из листинга (рис. 4.2).

```
avbogdanyuk@Bogdanyuk: ~/work/arch-pc/lab10
GNU nano 6.2 lab10-1.asm *
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`)
mov ebx, esi
mov eax, 6
int 80h
call quit_
^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute   ^C
^X Exit      ^R Read File  ^\ Replace    ^U Paste      ^J Justify   ^/
```

Рис. 4.2: Файл lab10-1.asm

Листинг:

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

```

Создаю исполняемый файл и запускаю его. Проверяю с помощью ls и cat. Программа выполняется корректно (рис. 4.3).

```

avbogdanyuk@Bogdanyuk:~/work/arch-pc/lab10$ nasm -f elf -g -l lab10-1.lst lab10-1.asm
avbogdanyuk@Bogdanyuk:~/work/arch-pc/lab10$ ld -m elf_i386 -o lab10-1 lab10-1.o
avbogdanyuk@Bogdanyuk:~/work/arch-pc/lab10$ ./lab10-1
Введите строку для записи в файл: Hello world!
avbogdanyuk@Bogdanyuk:~/work/arch-pc/lab10$ ls -l
total 44
-rw-r--r-- 1 avbogdanyuk avbogdanyuk 3942 Nov  8 15:45 in_out.asm
-rwxr-xr-x 1 avbogdanyuk avbogdanyuk 9700 Dec 16 15:46 lab10-1
-rw-r--r-- 1 avbogdanyuk avbogdanyuk 1142 Dec 16 15:46 lab10-1.asm
-rw-r--r-- 1 avbogdanyuk avbogdanyuk 13448 Dec 16 15:46 lab10-1.lst
-rw-r--r-- 1 avbogdanyuk avbogdanyuk 2512 Dec 16 15:46 lab10-1.o
-rw-r--r-- 1 avbogdanyuk avbogdanyuk 13 Dec 16 15:46 readme-1.txt
-rw-r--r-- 1 avbogdanyuk avbogdanyuk 0 Dec 16 15:40 readme-2.txt
avbogdanyuk@Bogdanyuk:~/work/arch-pc/lab10$ cat readme-1.txt
Hello world!
avbogdanyuk@Bogdanyuk:~/work/arch-pc/lab10$

```

Рис. 4.3: Создание, запуск и проверка

С помощью команды `chmod` изменяю права доступа к исполняемому файлу `lab10-1` (а-х - для всех пользователей отменить право на исполнение), запрети его исполнение для всех. Пытаюсь выполнить его. Выводится сообщение 'Permission denied', потому что мне, как owner, запрещено исполнять файл (рис. 4.4).

```

avbogdanyuk@Bogdanyuk:~/work/arch-pc/lab10$ chmod a-x lab10-1
avbogdanyuk@Bogdanyuk:~/work/arch-pc/lab10$ ./lab10-1
-bash: ./lab10-1: Permission denied
avbogdanyuk@Bogdanyuk:~/work/arch-pc/lab10$

```

Рис. 4.4: Изменение права доступа, проверка

С помощью `chmod` изменяю право доступа к файлу `lab10-1.asm` с исходным текстом программы, добавив права на исполнение (а+х - для всех пользователей добавить право на исполнение) (рис. 4.5).

```

avbogdanyuk@Bogdanyuk:~/work/arch-pc/lab10$ chmod a+x lab10-1.asm

```

Рис. 4.5: Изменение права доступа lab10-1.asm

Пытаюсь выполнить `lab10-1.asm`. Выводится построчно содержимое файла (рис. 4.6).

```

avbogdanyuk@Bogdanyuk:~/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`'
avbogdanyuk@Bogdanyuk:~/work/arch-pc/lab10$

```

Рис. 4.6: Запуск lab10-1.asm

Мой вариант - 4. Сначала изменяю права доступа к файлу readme-1.txt в символьном виде. Проверяю с помощью ls, все сработало корректно (рис. 4.7).

```

avbogdanyuk@Bogdanyuk:~/work/arch-pc/lab10$ chmod uo-rx readme-1.txt
avbogdanyuk@Bogdanyuk:~/work/arch-pc/lab10$ chmod g-rwx readme-1.txt
avbogdanyuk@Bogdanyuk:~/work/arch-pc/lab10$ chmod uo+w readme-1.txt
avbogdanyuk@Bogdanyuk:~/work/arch-pc/lab10$ ls -l
total 44
-rw-r--r-- 1 avbogdanyuk avbogdanyuk 3942 Nov  8 15:45 in_out.asm
-rw-rw-rw- 1 avbogdanyuk avbogdanyuk 9700 Dec 16 15:46 lab10-1
-rwxr-xr-x 1 avbogdanyuk avbogdanyuk 1142 Dec 16 15:46 lab10-1.asm
-rw-r--r-- 1 avbogdanyuk avbogdanyuk 13448 Dec 16 15:46 lab10-1.lst
-rw-r--r-- 1 avbogdanyuk avbogdanyuk 2512 Dec 16 15:46 lab10-1.o
--w----w- 1 avbogdanyuk avbogdanyuk 13 Dec 16 15:46 readme-1.txt
-rw-r--r-- 1 avbogdanyuk avbogdanyuk 0 Dec 16 15:40 readme-2.txt
avbogdanyuk@Bogdanyuk:~/work/arch-pc/lab10$

```

Рис. 4.7: Изменение права доступа в символьном виде

Теперь изменяю права доступа к файлу readme-2.txt двоичном виде. 001 - 1, 011 - 3, 110 - 6. Проверяю с помощью ls, работает корректно (рис. 4.8).

```

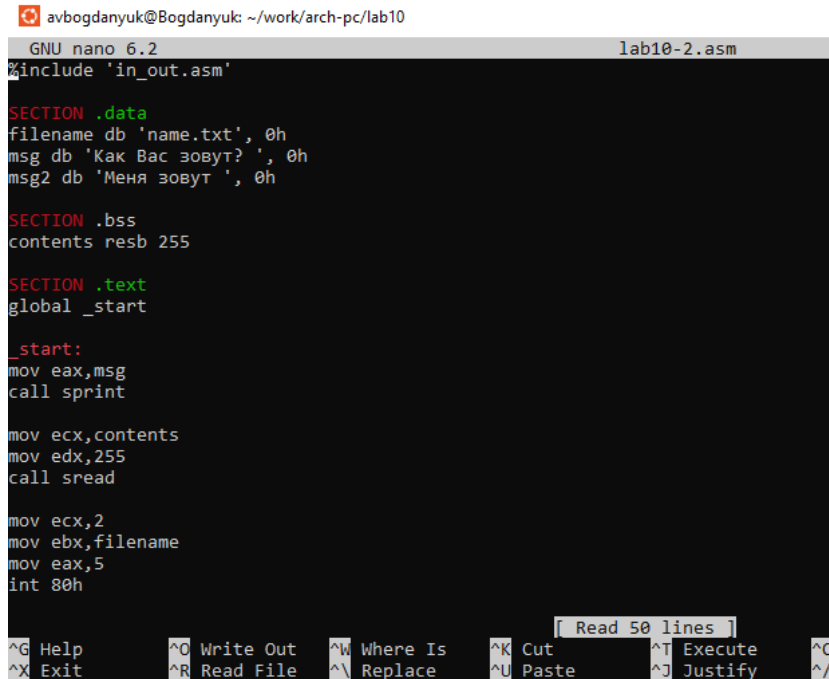
avbogdanyuk@Bogdanyuk:~/work/arch-pc/lab10$ chmod 136 readme-2.txt
avbogdanyuk@Bogdanyuk:~/work/arch-pc/lab10$ ls -l
total 44
-rw-r--r-- 1 avbogdanyuk avbogdanyuk 3942 Nov  8 15:45 in_out.asm
-rw-rw-rw- 1 avbogdanyuk avbogdanyuk 9700 Dec 16 15:46 lab10-1
-rwxr-xr-x 1 avbogdanyuk avbogdanyuk 1142 Dec 16 15:46 lab10-1.asm
-rw-r--r-- 1 avbogdanyuk avbogdanyuk 13448 Dec 16 15:46 lab10-1.lst
-rw-r--r-- 1 avbogdanyuk avbogdanyuk 2512 Dec 16 15:46 lab10-1.o
--w----w- 1 avbogdanyuk avbogdanyuk 13 Dec 16 15:46 readme-1.txt
---x-wxrw- 1 avbogdanyuk avbogdanyuk 0 Dec 16 15:40 readme-2.txt
avbogdanyuk@Bogdanyuk:~/work/arch-pc/lab10$

```

Рис. 4.8: Изменение права доступа в двоичном виде

2. Задания для самостоятельной работы

Создаю файл lab10-2.asm, ввожу текст программы, чтобы: вывод приглашения “Как Вас зовут?”, ввести с клавиатуры свои фамилию и имя, создать файл с именем name.txt, записать в файл сообщение “Меня зовут”, дописать в файл строку введенную с клавиатуры, закрыть файл. (рис. 4.9).



```
GNU nano 6.2 lab10-2.asm
#include 'in_out.asm'

SECTION .data
filename db 'name.txt', 0h
msg db 'Как Вас зовут? ', 0h
msg2 db 'Меня зовут ', 0h

SECTION .bss
contents resb 255

SECTION .text
global _start

_start:
mov eax,msg
call sprint

mov ecx,contents
mov edx,255
call sread

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

^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute
^X Exit      ^R Read File ^_ Replace   ^U Paste     ^J Justify
^_/
```

Рис. 4.9: Файл lab10-2.asm

Листинг:

```
%include 'in_out.asm'

SECTION .data
filename db 'name.txt', 0h
msg db 'Как Вас зовут? ', 0h
msg2 db 'Меня зовут ', 0h
```

```
SECTION .bss
contents resb 255
```

```
SECTION .text
global _start
```

```
_start:
```

```
mov eax,msg
```

```
call sprint
```

```
mov ecx,contents
```

```
mov edx,255
```

```
call sread
```

```
mov ecx,2
```

```
mov ebx,filename
```

```
mov eax,5
```

```
int 80h
```

```
mov esi,eax
```

```
mov eax,msg2
```

```
call slen
```

```
mov edx, eax
```

```
mov ecx, msg2
```

```
mov ebx, esi
```

```
mov eax, 4
```

```
int 80h
```

```
mov eax, contents
call slen
```

```
mov edx, eax
mov ecx, contents
mov ebx, esi
mov eax, 4
int 80h
```

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

Создаю исполняемый файл и запускаю его, проверяю с помощью cat. Работает корректно (рис. 4.10).

```
avbogdanyuk@Bogdanyuk:~/work/arch-pc/lab10$ nasm -f elf -g -l lab10-2.lst lab10-2.asm
avbogdanyuk@Bogdanyuk:~/work/arch-pc/lab10$ ld -m elf_i386 -o lab10-2 lab10-2.o
avbogdanyuk@Bogdanyuk:~/work/arch-pc/lab10$ ./lab10-2
Как Вас зовут? Богданюк Анна
avbogdanyuk@Bogdanyuk:~/work/arch-pc/lab10$ cat name.txt
Меня зовут Богданюк Анна
```

Рис. 4.10: Создание и запуск

Проверяю с помощью ls (рис. 4.11).

```
avbogdanyuk@Bogdanyuk:~/work/arch-pc/lab10$ ls
in_out.asm  lab10-1.asm  lab10-1.o  lab10-2.asm  lab10-2.o  readme-1.txt
lab10-1     lab10-1.lst  lab10-2   lab10-2.lst  name.txt   readme-2.txt
avbogdanyuk@Bogdanyuk:~/work/arch-pc/lab10$
```

Рис. 4.11: Проверка

5 Выводы

В ходе выполнения лабораторной работы были приобретены навыки написания программ для работы с файлами.

Список литературы