

Отчёт по лабораторной работе №6

**Работа с Midnight Commander. Структура программы на языке
ассемблера NASM. Системные вызовы в ОС GNU Linux**

Мулин Иван Владимирович

Содержание

1	Цель работы	4
2	Задание	5
3	Ход работы	6
3.1	Выполнение лабораторной работы	6
3.2	Выполнение заданий для самостоятельной работы	9
4	Листинги написанных программ	10
5	Заключение	15

Список иллюстраций

3.1	Каталог lab06	6
3.2	Проверка работы программы lab6-1	7
3.3	Копирование файла in_out	7
3.4	Копирование файла lab6-1.asm	8
3.5	Проверка работы lab6-2	8
3.6	Изменённая программа lab6-2	9
3.7	Запуск программы lab6-1a	9
3.8	Запуск программы lab6-2a	9

1 Цель работы

Цель выполнения лабораторной работы № 6 заключается в освоении работы с файловым менеджером Midnight Commander и командами ассемблера `mov` и `int`. По адресу https://github.com/ivmulin/study_2022-2023_arch-rc находится репозиторий автора.

2 Задание

В файловом менеджере Midnight Commander написать и скомпилировать программы на языке ассемблера NASM.

3 Ход работы

3.1 Выполнение лабораторной работы

Открыв программу Midnight Commander, перейдём в рабочую директорию, создадим каталог lab06:

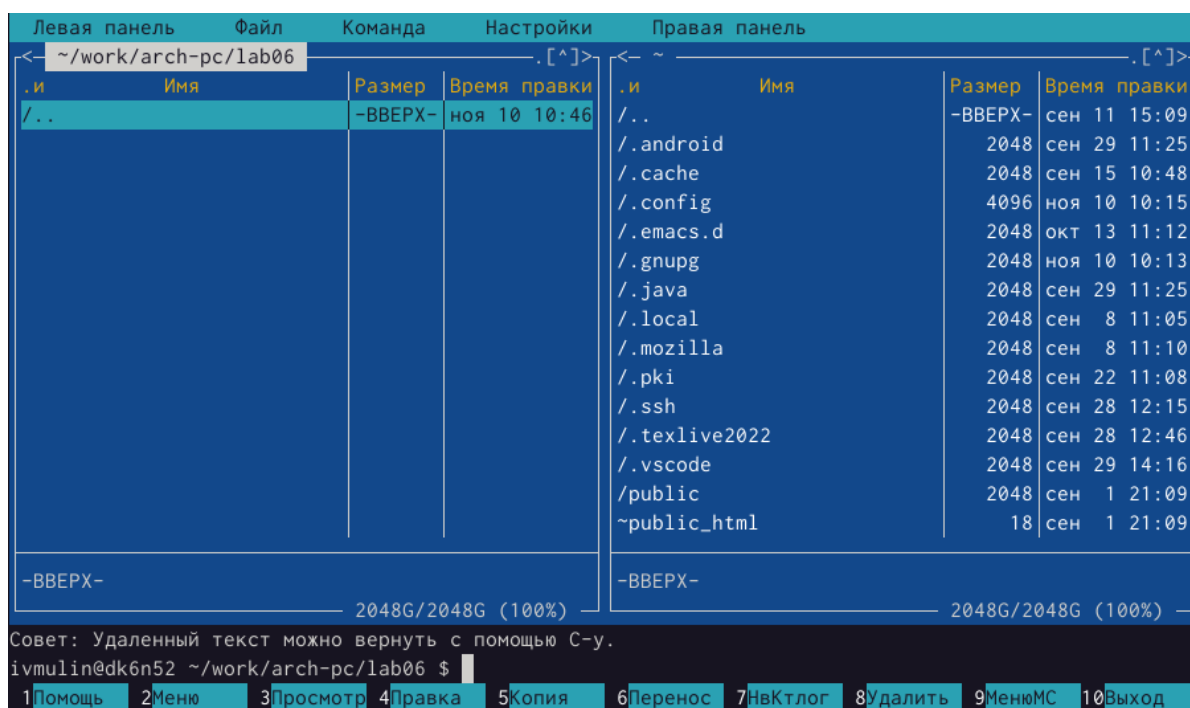


Рис. 3.1: Каталог lab06

Компилируем написанную программу lab6-1.asm и проверяем её работу:

```

ivmulin@dk6n52 ~/work/arch-pc/lab06 $ nasm -f elf lab6-1.asm
ivmulin@dk6n52 ~/work/arch-pc/lab06 $ ld -m elf_i386 -o lab6-1 lab6-1.o
ivmulin@dk6n52 ~/work/arch-pc/lab06 $ ./lab6-1
Введите строку
Мулин Иван Владимирович

```

Рис. 3.2: Проверка работы программы lab6-1

Используя интерфейс программы Midnight Commander, копируем файл `in_out.asm` в рабочую директорию:

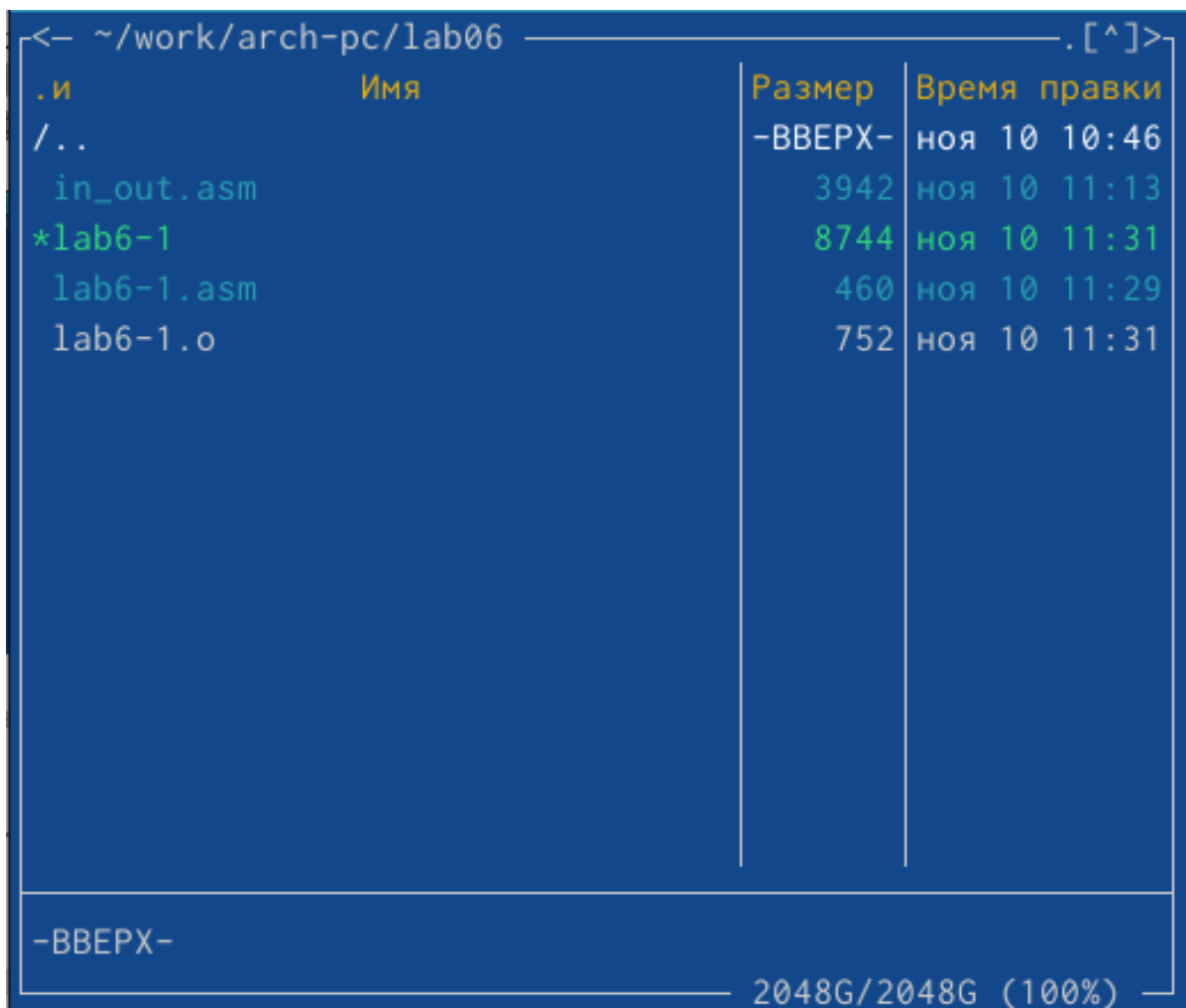


Рис. 3.3: Копирование файла in_out

Скопируем файл `lab6-1.asm`:

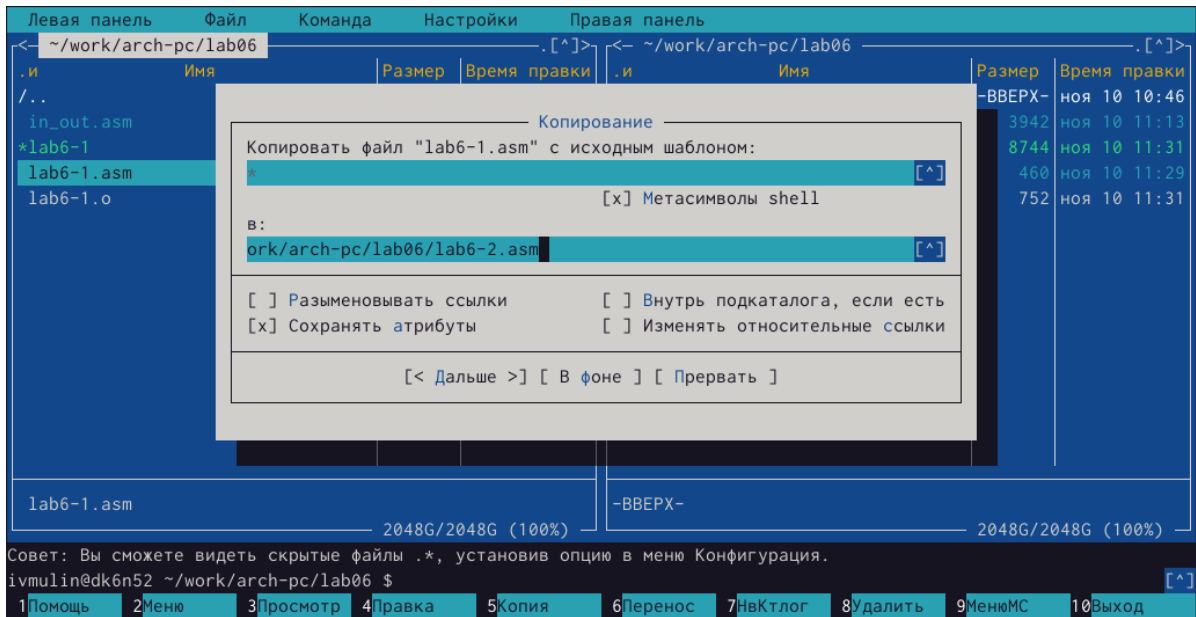


Рис. 3.4: Копирование файла lab6-1.asm

Запустим написанную программу lab6-2, использующую функции из файла in_out.asm:

```
ivmulin@dk8n54 ~/work/arch-pc/lab06 $ ./lab6-2
Введите строку

Мулин Иван Владимирович
ivmulin@dk8n54 ~/work/arch-pc/lab06 $
```

Рис. 3.5: Проверка работы lab6-2

В коде программы lab6-2 содержится вызов команды `sprintf`, выводящей, кроме сообщения, символ перевода строки:

```
; write
mov eax, message
call sprintf
```

Если вместо неё использовать команду `printf`, то программа будет выводить строку без символа перевода строки:


```
ivmulin@dk8n54 ~/work/arch-pc/lab06 $ ./lab6-2
Введите строку
Мулин Иван Владимирович
ivmulin@dk8n54 ~/work/arch-pc/lab06 $
```

Рис. 3.6: Изменённая программа lab6-2

3.2 Выполнение заданий для самостоятельной работы

Копию lab6-1a.asm написанной программы lab6-1.asm изменим так, что программа, помимо прочего, выводит полученную строку:

```
ivmulin@dk8n54 ~/work/arch-pc/lab06 $ ./lab6-1a
Введите строку
Мулин
Мулин
```

Рис. 3.7: Запуск программы lab6-1a

Аналогичные действия выполним в файле lab6-2a.asm:

```
ivmulin@dk8n54 ~/work/arch-pc/lab06 $ ./lab6-2a
Введите строку
Мулин
Мулин
```

Рис. 3.8: Запуск программы lab6-2a

4 Листинги написанных программ

Ввиду того, что ТУИС РУДН не разрешает отправлять в качестве отчёта более шести файлов, только программы lab6-1a.asm и lab6-2a.asm будут прикреплёны.

1. lab6-1.asm

```
section .data
```

```
message: db 'Введите строку', 10
```

```
length: equ $-message
```

```
section .bss
```

```
buf1: resb 80
```

```
section .text
```

```
global _start
```

```
_start:
```

```
    ; write
```

```
    mov eax, 4
```

```
    mov ebx, 1
```

```
    mov ecx, message
```

```
    mov edx, length
```

```
    int 80h ; вызов ядра
```

```

; read
mov eax, 3
mov ebx, 0
mov ecx, buf1
mov edx, 80 ; длина вводимой строки
int 80h

```

```

; exit
mov eax, 1
mov ebx, 0
int 80h

```

2. lab6-2.asm

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

section .data

```

message: db 'Введите строку', 10
length: equ $-message

```

section .bss

```
buf1: resb 80
```

section .text

```
global _start
```

```
_start:
```

```

; write
mov eax, message
call sprint

```

```

; read

```

```
mov ecx, buf1
mov edx, 80
call sread
```

```
; exit
call quit
```

3. lab6-1a.asm

section .data

```
message: db 'Введите строку', 10
length: equ $-message
```

section .bss

```
buf1: resb 80
```

section .text

global _start

_start:

```
; write
mov eax, 4
mov ebx, 1
mov ecx, message
mov edx, length
int 80h ; вызов ядра
```

```
; read
mov eax, 3
mov ebx, 0
mov ecx, buf1
mov edx, 80 ; длина вводимой строки
```

```

int 80h

; write
mov eax, 4
mov ebx, 1
mov ecx, buf1
mov edx, 80
int 80h

; exit
mov eax, 1
mov ebx, 0
int 80h

```

4. lab6-2a.asm

```

#include 'in_out.asm'

section .data
    message: db 'Введите строку', 10
    length: equ $-message

section .bss
    buf1: resb 80

section .text
global _start
_start:
    ; write
    mov eax, message
    call sprint

```

```
; read
mov ecx, buf1
mov edx, 80
call sread

; write
mov eax, buf1
call sprint

; exit
call quit
```

5 Заключение

В ходе выполнения данной лабораторной работы был изучен процесс работы в Midnight Commander и написаны программы, использующие изученные программы `mov` и `int`. Навык отработан в ходе выполнения заданий для самостоятельной работы.

Цель лабораторной работы, таким образом, была достигнута.