

Отчет по лабораторной работе №5

Дисциплина: архитектура компьютера

Бородин Дмитрий Алексеевич

Содержание

1	Цель работы	1
2	Задание	1
3	Теоретическое введение	1
4	Выполнение лабораторной работы	2
4.1	Основы работы с Midnight Commander	2
4.2	Работа в NASM	6
4.3	Подключение внешнего файла	9
4.4	Задание для самостоятельной работы	14
5	Выводы.....	19
	Список литературы.....	19

1 Цель работы

Целью данной лабораторной работы является приобретение практических навыков работы в Midnight Commander, освоение инструкций языка ассемблера mov и int.

2 Задание

1. Основы работы с mc
2. Структура программы на языке ассемблера NASM
3. Подключение внешнего файла
4. Выполнение заданий для самостоятельной работы

3 Теоретическое введение

Midnight Commander (или просто mc) — это программа, которая позволяет просматривать структуру каталогов и выполнять основные операции по управлению файловой системой, т.е. mc является файловым менеджером. Midnight Commander позволяет сделать работу с файлами более удобной и наглядной. Программа на языке ассемблера NASM, как правило, состоит из трёх секций: секция кода программы (SECTION .text), секция инициализированных (известных во время компиляции) данных (SECTION .data) и секция неинициализированных данных (тех, под которые во время компиляции только отводится память, а значение

присваивается в ходе выполнения программы) (SECTION .bss). Для объявления инициализированных данных в секции .data используются директивы DB, DW, DD, DQ и DT, которые резервируют память и указывают, какие значения должны храниться в этой памяти: - DB (define byte) — определяет переменную размером в 1 байт; - DW (define word) — определяет переменную размером в 2 байта (слово); - DD (define double word) — определяет переменную размером в 4 байта (двойное слово); - DQ (define quad word) — определяет переменную размером в 8 байт (четверёхбайтное слово); - DT (define ten bytes) — определяет переменную размером в 10 байт. Директивы используются для объявления простых переменных и для объявления массивов. Для определения строк принято использовать директиву DB в связи с особенностями хранения данных в оперативной памяти. Инструкция языка ассемблера mov предназначена для дублирования данных источника в приёмнике.

```
mov dst,src
```

Здесь operand dst — приёмник, а src — источник. В качестве операнда могут выступать регистры (register), ячейки памяти (memory) и непосредственные значения (const). Инструкция языка ассемблера int предназначена для вызова прерывания с указанным номером.

```
int n
```

Здесь n — номер прерывания, принадлежащий диапазону 0–255. При программировании в Linux с использованием вызовов ядра sys_calls n=80h (принято задавать в шестнадцатеричной системе счисления).

4 Выполнение лабораторной работы

4.1 Основы работы с Midnight Commander

Введя соответствующую команду в терминале (рис. 1), я открываю Midnight Commander (рис. 2).

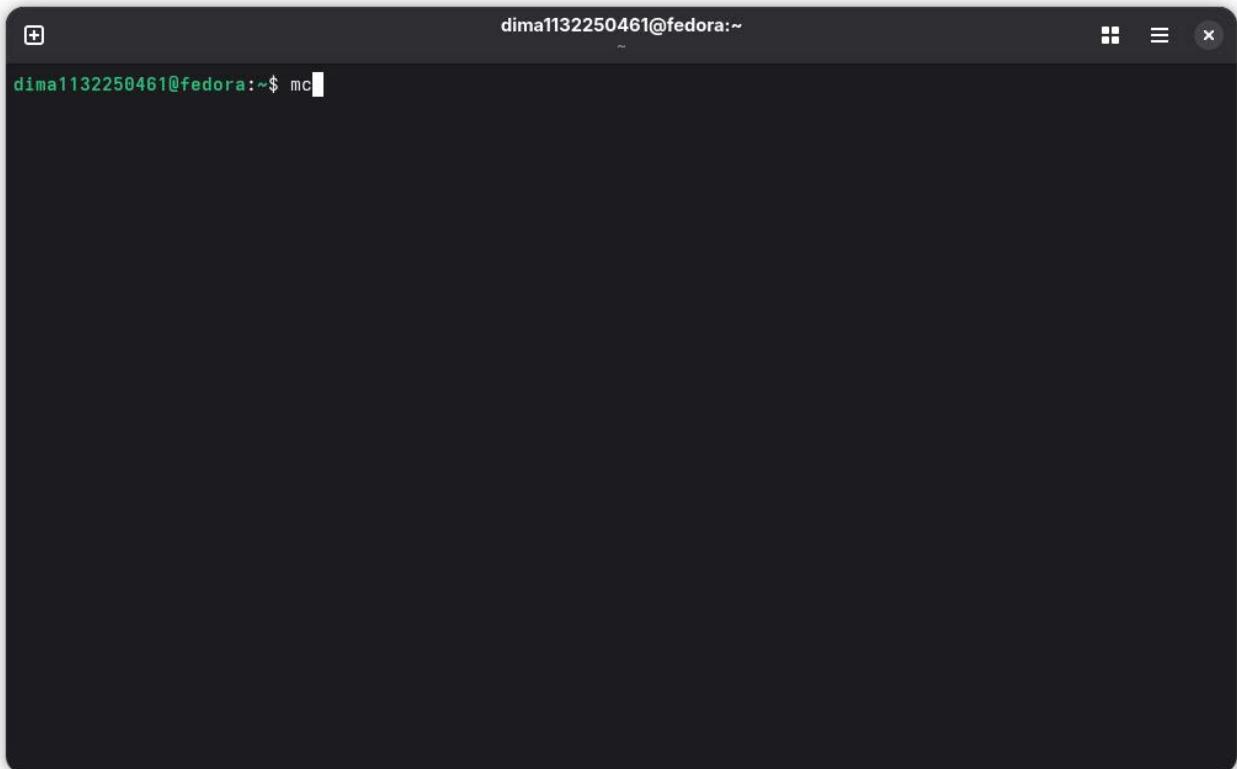


Рис. 1: Открытие Midnight Commander

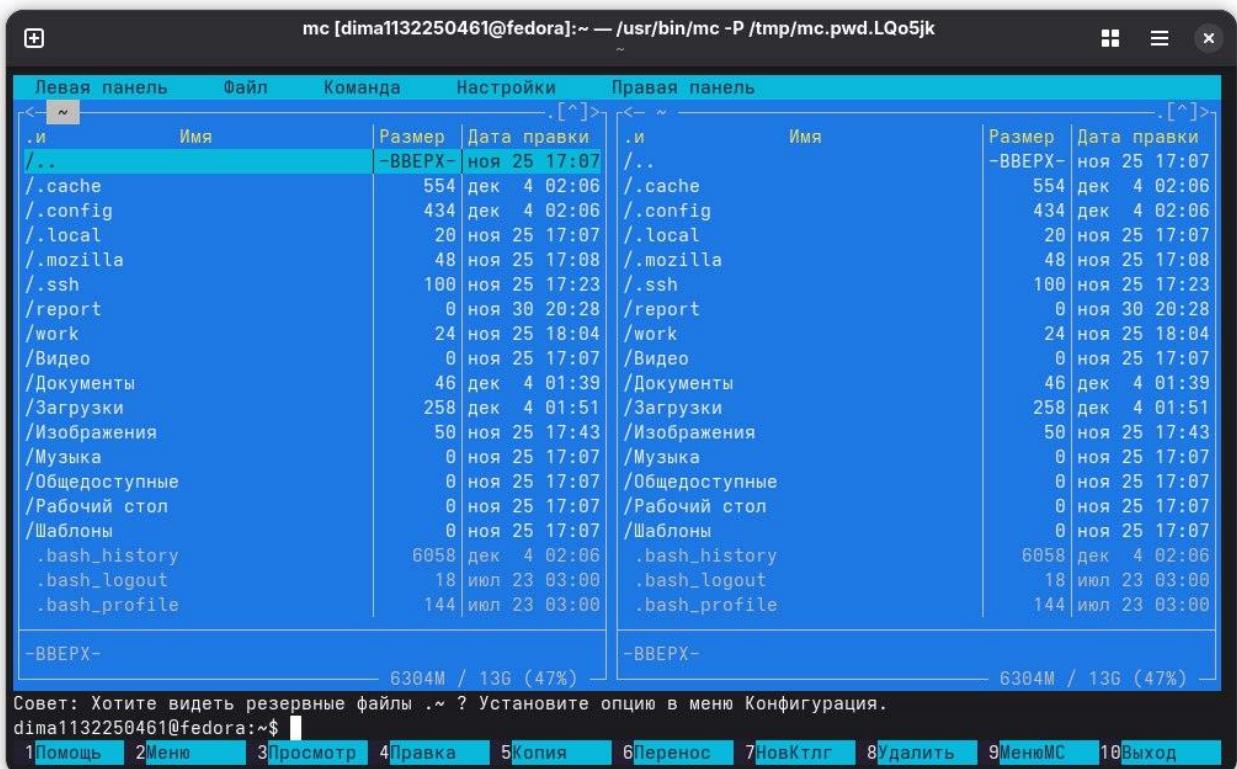


Рис. 2: Интерфейс Midnight Commander

Перехожу в созданный каталог в предыдущей лабораторной работе (рис. 3).

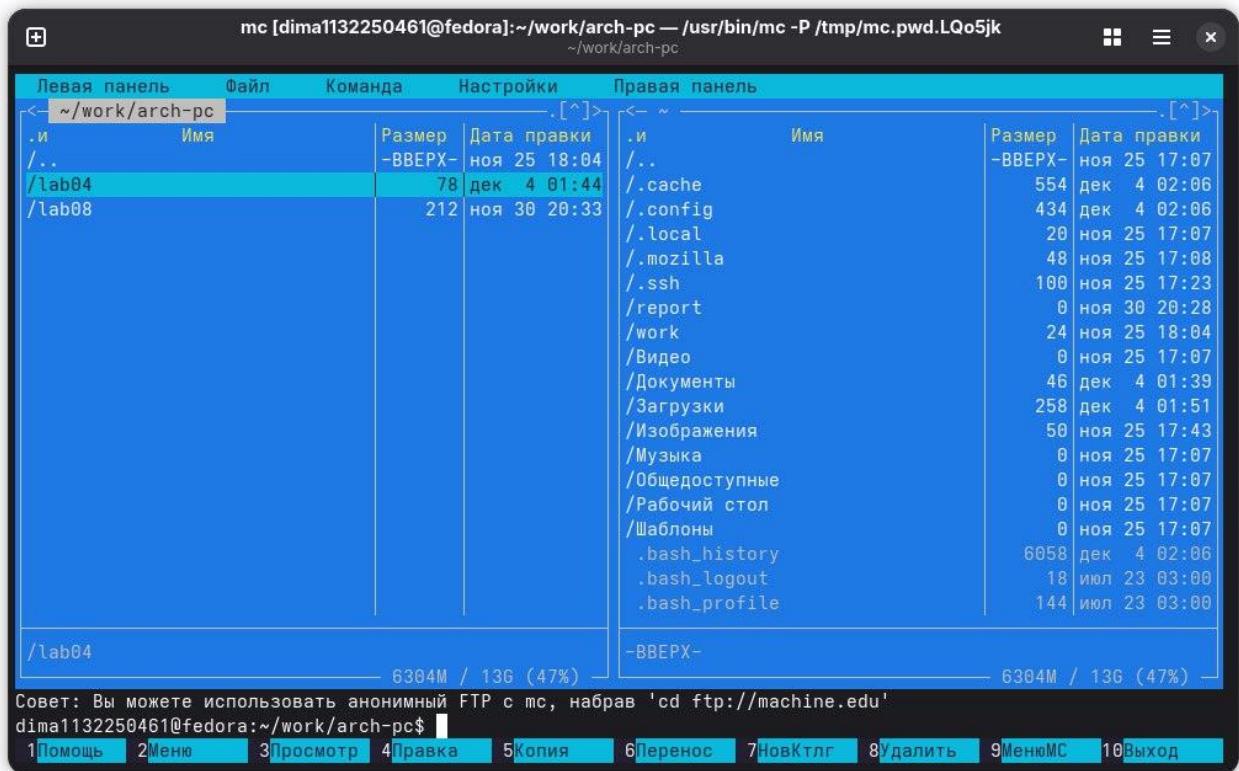


Рис. 3: Открытый каталог arch-pc

С помощью функциональной клавиши, я создаю подкаталог lab05, в котором буду работать (рис. 4).

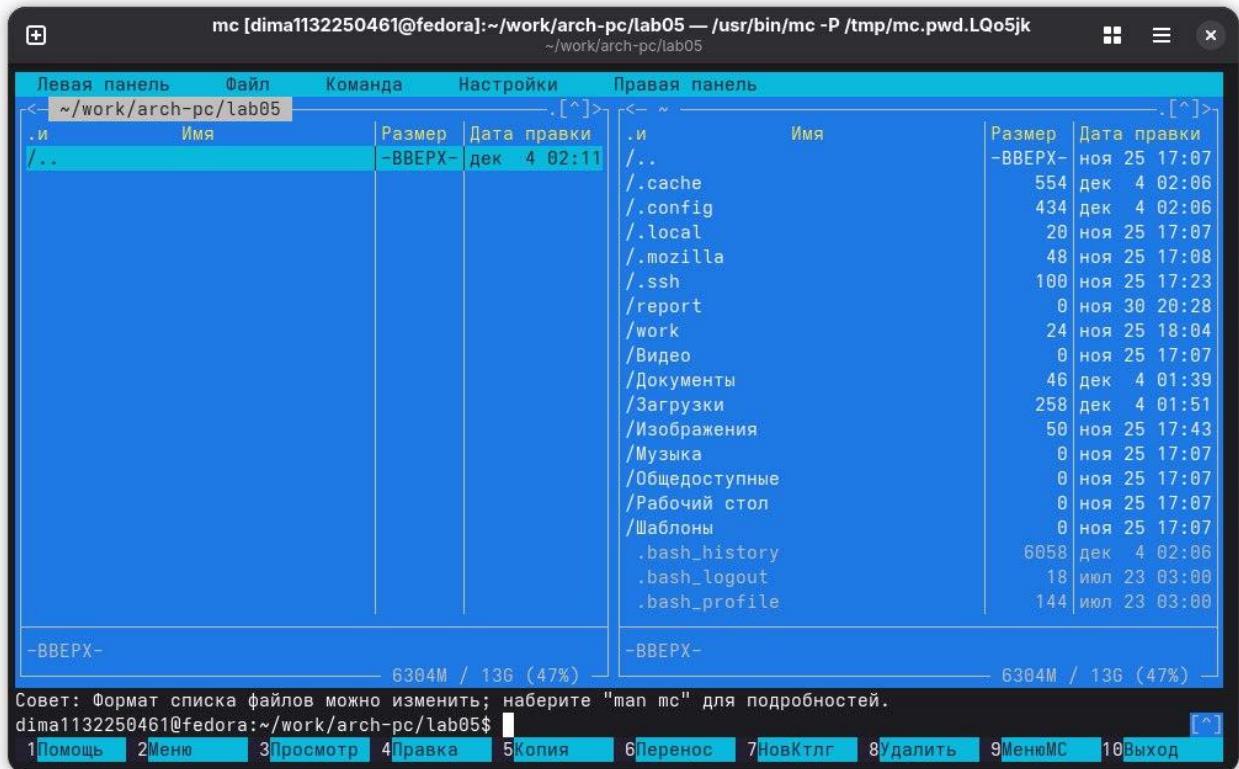


Рис. 4: Создание рабочего подкаталога

В строке ввода вводжу команду touch и создаю файл (рис. 5).

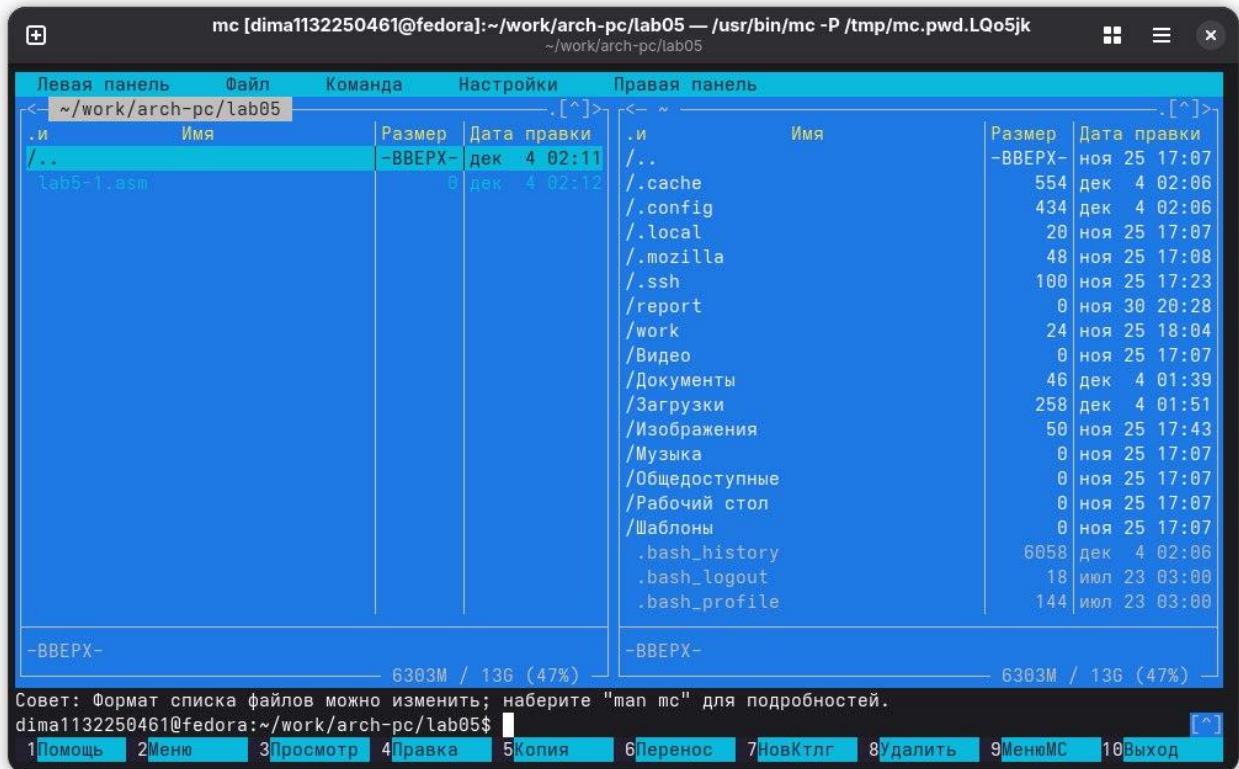


Рис. 5: Создание файла в *Midnight Commander*

4.2 Работа в NASM

С помощью F4 открываю только что созданный файл и вношу код с листинга (рис. 6).

The screenshot shows a terminal window titled 'mc [dima1132250461@fedora]:~/work/arch-pc/lab05 — /usr/bin/mc -P /tmp/mc.pwd.LQo5jk ~/work/arch-pc/lab05'. The window displays assembly code for a program named 'lab5-1.asm'. The code includes sections for .data, .bss, and .text, defining variables like msg and buf1, and implementing a main loop with system calls for reading and writing. The assembly instructions use registers eax, ebx, ecx, edx, and int 0x80 for system calls. The bottom of the window shows a menu bar with items 1 Помощь, 2 Сохран, 3 Блок, 4 Замена, 5 Копия, 6 Перенести, 7 Поиск, 8 Удалить, 9 МенюMC, and 10 Выход.

```
lab5-1.asm      [---]  0 L:[ 1+ 0   1/ 27 ] *(0   / 489b) 0083 0x053      [*][X]
SECTION .data
msg: DB 'Введите строку:', 10
msgLen: EQU $-msg

SECTION .bss
buf1: RESB 80

SECTION .text
GLOBAL _start

_start:
    mov    eax, 4
    mov    ebx, 1
    mov    ecx, msg
    mov    edx, msgLen
    int    0x80

    mov    eax, 3
    mov    ebx, 0
    mov    ecx, buf1
    mov    edx, 80
    int    0x80

    mov    eax, 1
    mov    ebx, 0
```

Рис. 6: Редактирование файла в Midnight Commander

Проверяю сохраненные изменения с помощью клавиши F3 (рис. 7).

The screenshot shows a terminal window titled 'mc [dima1132250461@fedora]:~/work/arch-pc/lab05 — /usr/bin/mc -P /tmp/mc.pwd.LQo5jk ~/work/arch-pc/lab05'. The main area displays the content of the assembly file 'lab5-1.asm'. The code defines a global variable 'msg' containing the string 'Введите строку:', a label 'buf1' for a 80-byte buffer, and the entry point '_start' which reads the string into memory and prints it. The bottom of the window shows a menu bar with items 1 through 10, where item 2 'Сохран' (Save) is highlighted.

```
lab5-1.asm      [---]  0 L:[ 1+ 0   1/ 27] *(0   / 489b) 0083 0x053      [*][X]
SECTION .data
msg: DB 'Введите строку:', 10
msgLen: EQU $-msg

SECTION .bss
buf1: RESB 80

SECTION .text
GLOBAL _start

_start:
    mov    eax, 4......
    mov    ebx, 1......
    mov    ecx, msg......
    mov    edx, msgLen...
    int    0x80......

    mov    eax, 3......
    mov    ebx, 0......
    mov    ecx, buf1.....
    mov    edx, 80......
    int    0x80......

    mov    eax, 1......
    mov    ebx, 0......



1Помощь 2Сохран 3Блок 4Замена 5Копия 6Перенить 7Поиск 8Удалить 9МенюMC 10Выход
```

Рис. 7: Проверка сохранения сделанных изменений

Транслирую и компоную измененный файл, запускаю (рис. 8).

The screenshot shows a terminal window with two tabs. The left tab has the title "dima1132250461@fedora:~/work/arch-pc/lab05" and contains the command "nasm -f elf lab5-1.asm". The right tab also has the same title and contains the commands "ld -m elf_i386 -o lab5-1 lab5-1" and "./lab5-1". Below the tabs, the user types "Ведите строку:" followed by "Бородин Дмитрий Алексеевич" and presses Enter. The terminal prompt "dima1132250461@fedora:~/work/arch-pc/lab05 \$" is visible at the bottom.

99-а уда

Рис. 8: Трансляция, компоновка и последующий запуск программы

4.3 Подключение внешнего файла

Скачанный с ТУИС файл сохраняю в общую папку на своем компьютере, на виртуальной машине в интерфейсе Midnight Commander перехожу в директорию общей папки, копирую файл в рабочий подкаталог. (рис. 9).

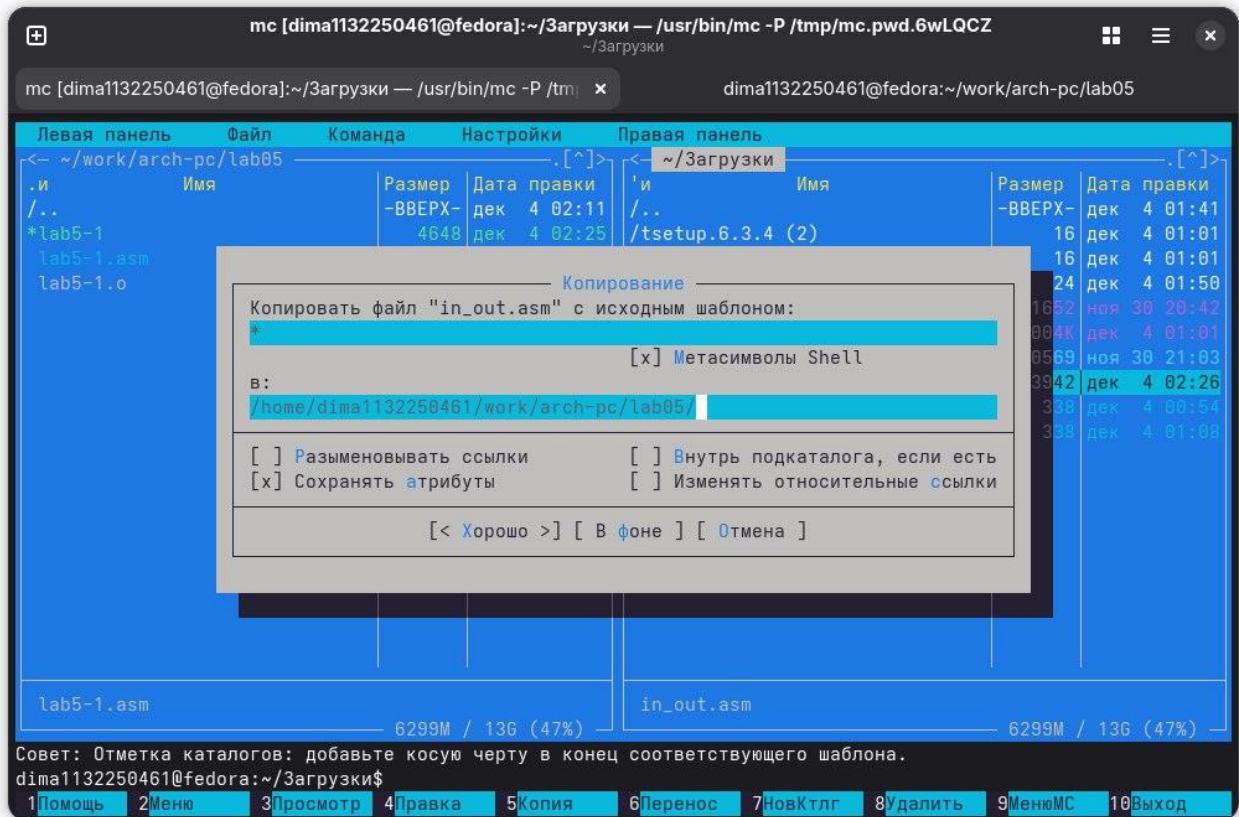


Рис. 9: Копирование файла в рабочий каталог

Создаю копию файла для последующей работы с ним (рис. 10).

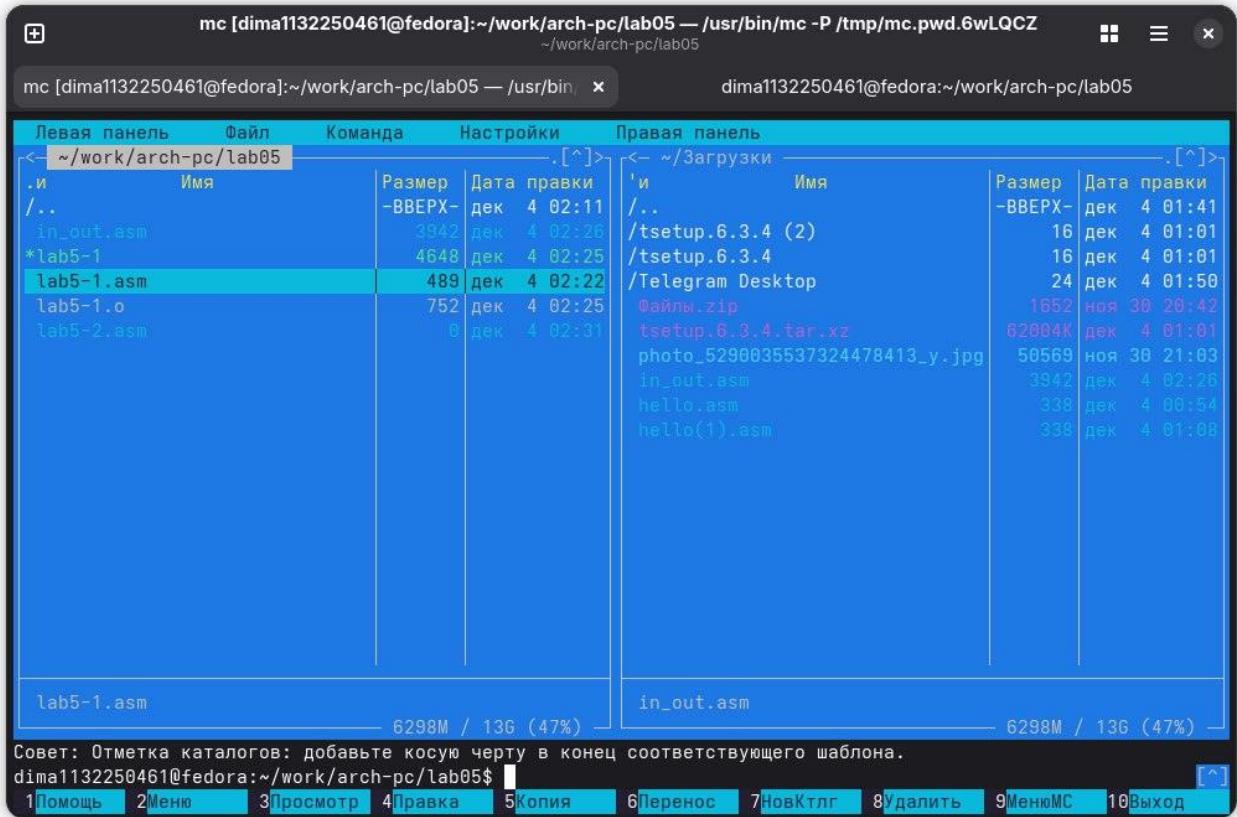


Рис. 10: Создание копии файла в Midnight Commander

В копии файла подключаю подпрограмм из подключенного файла (рис. 11).

The screenshot shows the Mars Cross-Assembler (mc) application window. The title bar indicates the user is at `dimma1132250461@fedora:~/work/arch-pc/lab05`. The status bar shows the file path `/home/dimma1132250461/work/arch-pc/lab05/lab5-2.asm`, progress `228/228`, and zoom level `100%`. The main area displays the assembly code:

```
%include 'in_out.asm'
SECTION .data
msg: DB 'Введите строку: ',0h
SECTION .bss
buf1: RESB 80
SECTION .text
GLOBAL _start
_start:
    mov eax, msg
    call sprintLF
    mov ecx, buf1
    mov edx, 80
    call sread
    call quit
```

The bottom of the window features a toolbar with buttons numbered 1 through 10, corresponding to various menu items: Помощь (Help), Разверн (Zoom), Выход (Exit), Нех (Neh), Перейти (Jump), Поиск (Search), Исходный (Source), Формат (Format), and Выход (Exit).

Рис. 11: Изменение программы

Транслирую, компоную и запускаю программу с подключенным файлом (рис. 12).

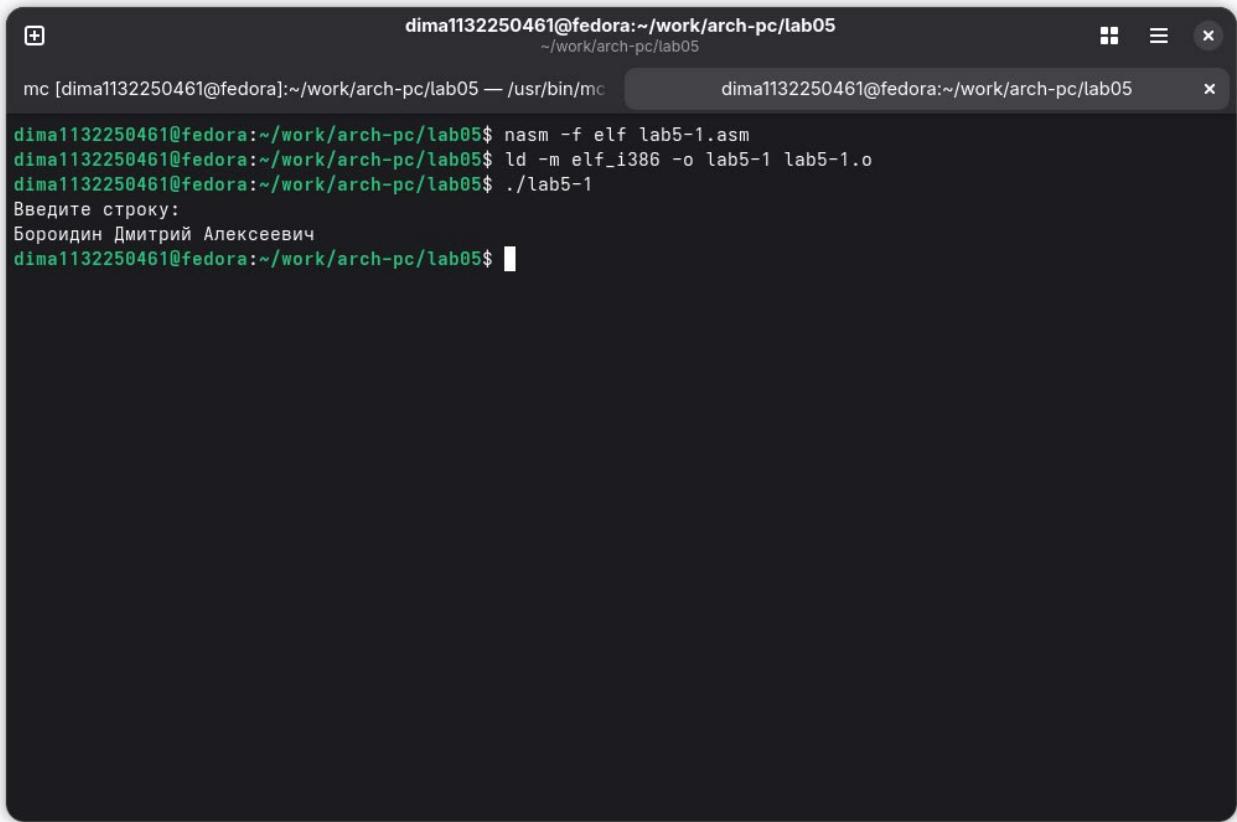
The screenshot shows a terminal window with a dark background and light-colored text. At the top, it displays the user's name, host, current directory, and the command prompt. The user has run several commands to compile an assembly file into an ELF executable and then run it. The output of the program is also visible in the terminal.

```
dima1132250461@fedora:~/work/arch-pc/lab05
~/work/arch-pc/lab05
dima1132250461@fedora:~/work/arch-pc/lab05$ nasm -f elf lab5-1.asm
dima1132250461@fedora:~/work/arch-pc/lab05$ ld -m elf_i386 -o lab5-1 lab5-1.o
dima1132250461@fedora:~/work/arch-pc/lab05$ ./lab5-1
Введите строку:
Бородин Дмитрий Алексеевич
dima1132250461@fedora:~/work/arch-pc/lab05$
```

Рис. 12: Запуск измененной программы

Редактирую файл и заменяю в нем подпрограмму sprintLF на sprint. Разница подпрограмм в том, что вторая вызывает ввод на той же строке (рис. 13).

4.4 Задание для самостоятельной работы



The screenshot shows a terminal window with a dark background and light-colored text. At the top, it displays the user's name and session information: `dima1132250461@fedora:~/work/arch-pc/lab05`. Below this, the terminal prompt is shown twice: `mc [dima1132250461@fedora]:~/work/arch-pc/lab05 — /usr/bin/mc` and `dima1132250461@fedora:~/work/arch-pc/lab05`. The terminal then shows the command history:

```
dima1132250461@fedora:~/work/arch-pc/lab05$ nasm -f elf lab5-1.asm
dima1132250461@fedora:~/work/arch-pc/lab05$ ld -m elf_i386 -o lab5-1 lab5-1.o
dima1132250461@fedora:~/work/arch-pc/lab05$ ./lab5-1
Введите строку:
Бородин Дмитрий Алексеевич
dima1132250461@fedora:~/work/arch-pc/lab05$
```

Рис. 13: Запуск измененной программы с другой подпрограммой

Создаю копию `lab5-1.asm`, редактирую так, чтобы в конце выводилась введенная мною строка с клавиатуры (рис. 14).

The screenshot shows a terminal window titled 'mc [dima1132250461@fedora]:~/work/arch-pc/lab05 — /usr/bin/mc -P /tmp/mc.pwd.6wLQCZ ~/work/arch-pc/lab05'. The window contains assembly code for a program named 'lab5-1copy.asm'. The code is written in AT&T syntax. It includes sections for .data, .bss, and .text. In the .text section, there is a label '_start' which contains code for printing a message and reading input. Below the assembly code, there is a menu bar with items 1 through 10, and a status bar at the bottom.

```
lab5-1copy.asm      [---] 7 L:[ 1+11 12/ 40 ] *(143 /1258b) 0010 0x00A      [*][X]
SECTION .data
msg: DB 'Введите строку:', 10
msgLen: EQU $ - msg

SECTION .bss
buf1: RESB 80

SECTION .text
GLOBAL _start

_start:
    ; Вывод приглашения
    mov    eax, 4          ; sys_write
    mov    ebx, 1          ; stdout
    mov    ecx, msg        ; указатель на строку
    mov    edx, msgLen    ; длина строки
    int    80h

    ; Чтение строки
    mov    eax, 3          ; sys_read
    mov    ebx, 0          ; stdin
    mov    ecx, buf1       ; буфер для ввода
    mov    edx, 80          ; размер буфера
    int    80h
```

Рис. 14: Редактирование копии

Транслирую, компоную и запускаю свою программу (рис. 15).

The screenshot shows a terminal window titled "dima1132250461@fedora:~/work/arch-pc/lab05". The user has run the command "nasm -f elf lab5-1copy.asm" to assemble the assembly code into an ELF file. Then, they used "ld -m elf_i386 -o lab5-1copy lab5-1copy.o" to link it. Finally, they executed the program with "./lab5-1copy". The terminal prompt "dima1132250461@fedora:~/work/arch-pc/lab05\$" is visible at the bottom.

```
dima1132250461@fedora:~/work/arch-pc/lab05$ nasm -f elf lab5-1copy.asm
dima1132250461@fedora:~/work/arch-pc/lab05$ ld -m elf_i386 -o lab5-1copy lab5-1copy.o
dima1132250461@fedora:~/work/arch-pc/lab05$ ./lab5-1copy
Ведите строку:
Бородин Дмитрий Алексеевич
Бородин Дмитрий Алексеевич
dima1132250461@fedora:~/work/arch-pc/lab05$
```

Рис. 15: Запуск своей программы

Код прикладываю

```
SECTION .data

msg: DB 'Введите строку:',10
msgLen: EQU $-msg

SECTION .bss
buf1: RESB 80

SECTION .text

GLOBAL _start

_start:
    mov    eax, 4
    mov    ebx, 1
    mov    ecx, msg
    mov    edx, msgLen
    int    80h
    mov    eax, 3
    mov    ebx, 0
    mov    ecx, buf1
    mov    edx, 80
```

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

удв

Создаю копию lab5-2.asm, редактирую так, чтобы в конце выводилась введеная мною строка с клавиатуры (рис. 16).

The screenshot shows the assembly code for `lab5-2copy.asm` in the mc (MCE) editor. The code is as follows:

```
SECTION .data

msg: DB 'Введите строку:',10
msgLen: EQU $-msg

SECTION .bss
buf1: RESB 80

SECTION .text

GLOBAL _start

_start:
    mov    eax, 4
    mov    ebx, 1
    mov    ecx, msg
    mov    edx, msgLen
    int    80h
    ...
    mov    eax, 3
    mov    ebx, 0
    mov    ecx, buf1
    mov    edx, 80
    int    80h
```

The menu bar at the bottom includes items 1 Помощь, 2 Сохран, 3 Блок, 4 Замена, 5 Копия, 6 Перенить, 7 Поиск, 8 Удалить, 9 МенюМС, and 10 Выход.

Рис. 16: Редактирование копии

Транслирую, компоную и запускаю свою программу (рис. 17).

The screenshot shows a terminal window with a dark background and light-colored text. The title bar reads "dima1132250461@fedora:~/work/arch-pc/lab05". The terminal has four tabs, all showing the same command-line interface. The visible portion of the terminal shows the following commands and their output:

```
dima1132250461@fedora:~/work/arch-pc/lab05$ nasm -f elf lab5-2copy.asm
dima1132250461@fedora:~/work/arch-pc/lab05$ ld -m elf_i386 -o lab5-2copy lab5-2copy.o
dima1132250461@fedora:~/work/arch-pc/lab05$ ./lab5-2copy
Ведите строку: Бородин Дмитрий Алексеевич
Бородин Дмитрий Алексеевич
dima1132250461@fedora:~/work/arch-pc/lab05$
```

Рис. 17: Запуск своей программы

Код прикладываю:

```
%include 'in_out.asm'

SECTION .data

msg: DB 'Введите строку: ', 0h
msgLen: EQU $-msg

SECTION .bss
buf1: RESB 80

SECTION .text
    GLOBAL _start
_start:

    mov eax, msg
    call sprint

    mov ecx, buf1
    mov edx, 80

    call sread
```

```
mov eax, 4
mov ebx, 1
mov ecx, buf1
int 80h

call quit
```

5 Выводы

При выполнении данной лабораторной работы я приобрёл практические навыки работы в Midnight Commander, а также освоил инструкции языка ассемблера `mov` и `int`.

Список литературы

1. Пример выполнения лабораторной работы
2. Курс на ТУИС
3. Лабораторная работа №5
4. Программирование на языке ассемблера NASM Столяров А. В.