

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ
Факультет физико-математических и естественных наук
Кафедра прикладной информатики и теории вероятностей

ОТЧЕТ
ПО ЛАБОРАТОРНОЙ РАБОТЕ №5
дисциплина: Архитектура компьютеров и операционные системы

Студент: Галиев Самир Салаватович

Группа: НКАбд-02-25

МОСКВА

2025 г.

Оглавление

- 1. Цель работы - стр. 3
- 2. Результаты выполнения лабораторной работы - стр. 3
 - 2.1 Работа с Midnight Commander - стр. 3
 - 2.2 Создание и выполнение первой программы - стр. 4
 - 2.3 Работа с внешним файлом in_out.asm - стр. 5
- 3. Задание для самостоятельной работы - стр. 7
 - 3.1 Модификация программы без in_out.asm - стр. 7
 - 3.1 Модификация программы с in_out.asm - стр. 8
- 4. Выводы - стр. 9
- 5. Список литературы - стр. 10

Цель работы

Приобретение практических навыков работы в Midnight Commander. Освоение инструкций языка ассемблера mov и int.

Результаты выполнения лабораторной работы

Работа с Midnight Commander

Задание: Изучение основ работы с файловым менеджером Midnight Commander, создание рабочего каталога и навигация по файловой системе.

Выполнение:

- 1) Запущен Midnight Commander командой mc
- 2) Создан каталог lab05 в директории ~/work/study/2025-2026/Архитектура компьютера/arch-pc с помощью клавиши F7
- 3) Осуществлена навигация по файловой системе с использованием клавиш ↑, ↓ и Enter

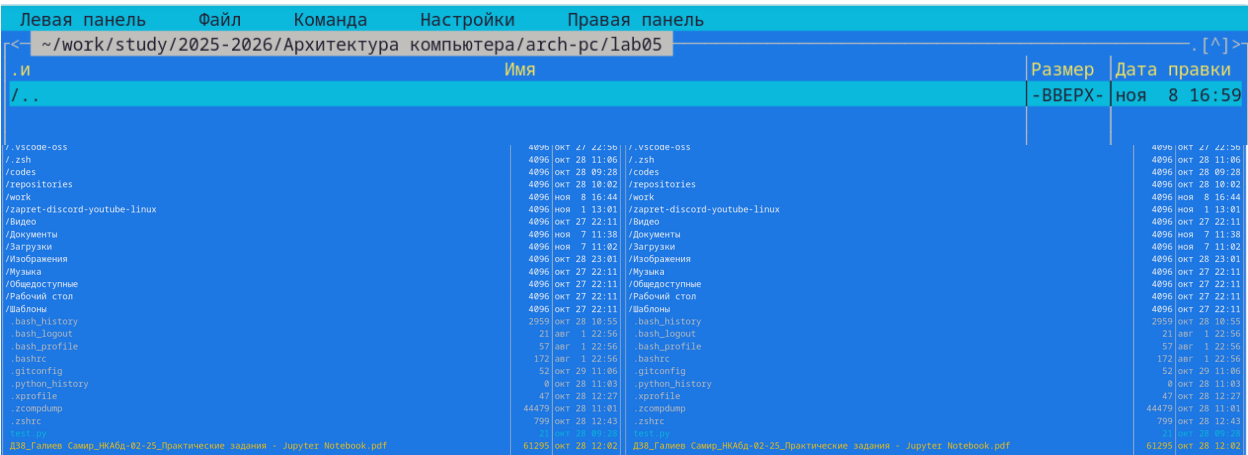


Рисунок 1: Окно Midnight Commander с созданным каталогом lab05

Комментарии: Midnight Commander предоставляет удобный графический интерфейс для работы с файловой системой, позволяя выполнять основные операции управления файлами с помощью функциональных клавиш.

Создание и выполнение первой программы

Задание: Создание программы на языке ассемблера NASM для вывода сообщения на экран и ввода строки с клавиатуры с использованием системных вызовов.

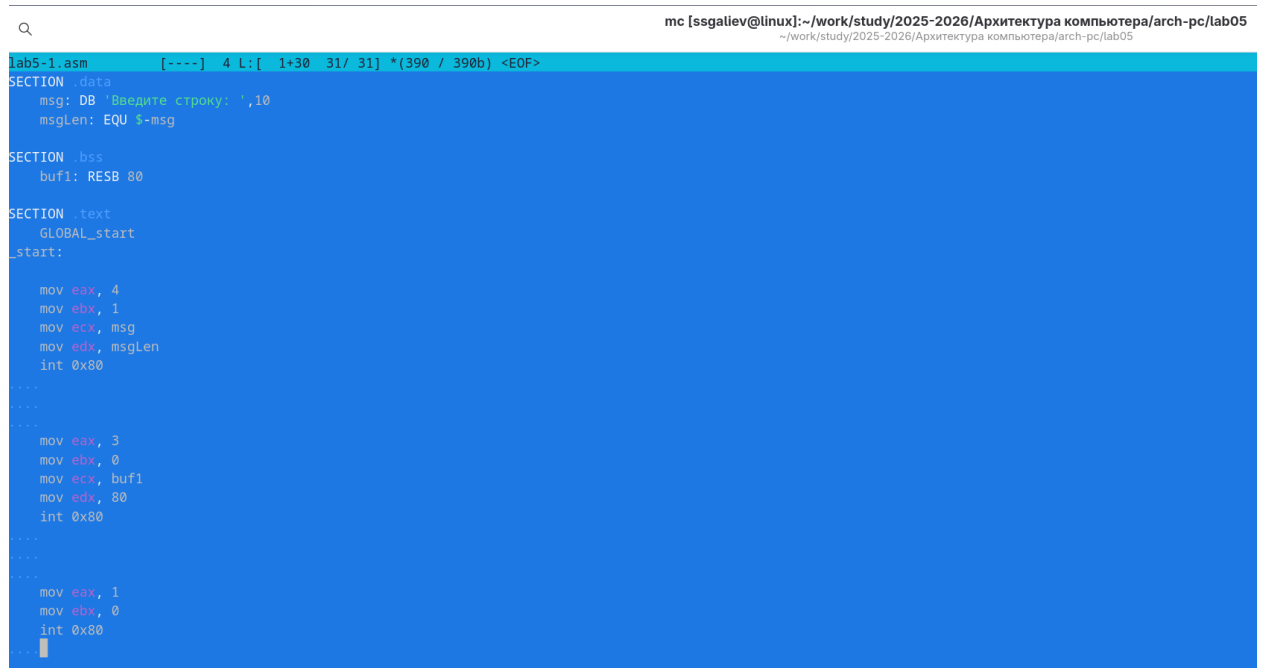
Выполнение:

- 1) Создан файл lab5-1.asm командой touch или через редактор nano
- 2) Написан код программы с использованием системных вызовов write, read и exit
- 3) Программа оттранслирована и скомпонована:

```
nasm -f elf32 lab5-1.asm
```

```
ld -m elf_i386 -o lab5-1 lab5-1.o
```

- 4) Программа запущена и протестирована



```
lab5-1.asm [----] 4 L: [ 1+30 31/ 31] *(390 / 390b) <EOF>
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

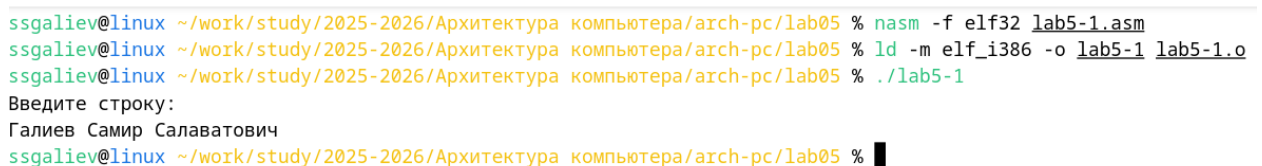
    ...

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

    ...

    mov eax, 1
    mov ebx, 0
    int 0x80
```

Рисунок 2: Код программы lab5-1.asm в редакторе nano



```
ssgaliev@linux ~/work/study/2025-2026/Архитектура компьютера/arch-pc/lab05 % nasm -f elf32 lab5-1.asm
ssgaliev@linux ~/work/study/2025-2026/Архитектура компьютера/arch-pc/lab05 % ld -m elf_i386 -o lab5-1 lab5-1.o
ssgaliev@linux ~/work/study/2025-2026/Архитектура компьютера/arch-pc/lab05 % ./lab5-1
Введите строку:
Галиев Самир Салаватович
ssgaliev@linux ~/work/study/2025-2026/Архитектура компьютера/arch-pc/lab05 % █
```

Рисунок 3: Трансляция, компоновка и выполнение программы lab5-1

Комментарии: Программа успешно выводит приглашение для ввода строки и ожидает пользовательский ввод. Использованы системные вызовы через инструкцию int 0x80 с передачей параметров через регистры.

Работа с внешним файлом in_out.asm

Задание: Подключение внешнего файла с подпрограммами и модификация программы для использования этих подпрограмм.

Выполнение:

- 1) Скачан файл in_out.asm и скопирован в рабочий каталог
- 2) Создана копия программы lab5-2.asm
- 3) Программа модифицирована для использования подпрограмм sprintLF, sread и quit
- 4) Проведено сравнение работы подпрограмм sprint и sprintLF

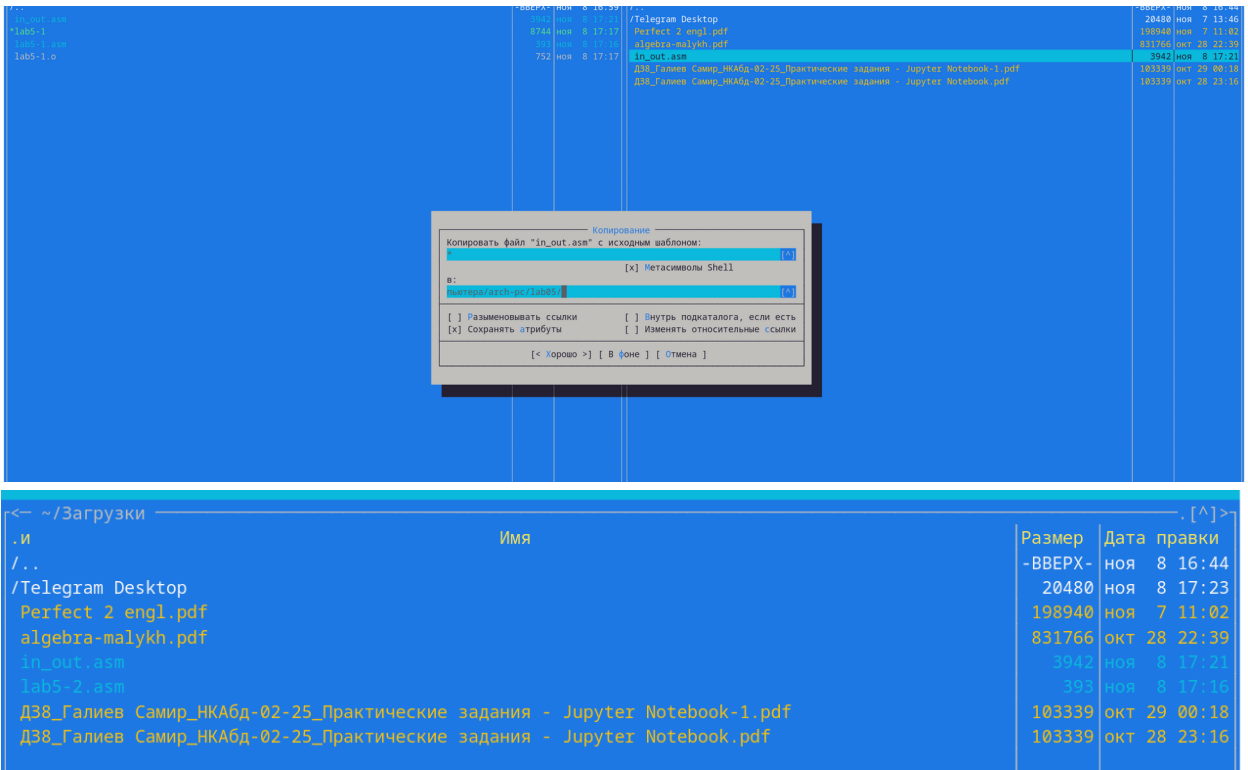


Рисунок 4: Копирование файла in_out.asm в рабочий каталог

```
ssgaliev@linux ~/Загрузки % nasm -f elf32 lab5-2.asm
ssgaliev@linux ~/Загрузки % ld -m elf_i386 -o lab5-2 lab5-2.o
ssgaliev@linux ~/Загрузки % ./lab5-2
```

Введите строку:

Галиев Самир Салаватович

```
ssgaliev@linux ~/Загрузки % nasm -f elf32 lab5-2.asm
ssgaliev@linux ~/Загрузки % ld -m elf_i386 -o lab5-2 lab5-2.o
ssgaliev@linux ~/Загрузки % ./lab5-2
```

Введите строку: Галиев Самир_Салаватович

Рисунок 5: Сравнение работы sprint и sprintLF

Комментарии: Подпрограмма sprintLF добавляет символ перевода строки после вывода сообщения, в то время как sprint выводит сообщение без перевода строки. Использование внешних подпрограмм упрощает написание кода и делает его более читаемым.

Задание для самостоятельной работы

Модификация программы без in_out.asm

Задание: Модификация программы lab5-1.asm для вывода введенной строки обратно на экран без использования внешнего файла in_out.asm.

Выполнение:

- 1) Создана копия файла lab5-3.asm
- 2) Добавлен системный вызов write после вызова read для вывода введенной строки
- 3) Программа собрана и протестирована

```
ssgaliev@linux ~/Загрузки % nasm -f elf32 lab5-3.asm
ssgaliev@linux ~/Загрузки % ld -m elf_i386 -o lab5-3 lab5-3.o
ssgaliev@linux ~/Загрузки % ./lab5-3
Введите строку:
Галиев Самри Салаватович
Галиев Самри Салаватович
ssgaliev@linux ~/Загрузки %
```

Рисунок 6: Работа программы lab5-3 с выводом введенной строки

Комментарии: Для вывода введенной строки использован системный вызов write с передачей адреса буфера и длины введенной строки. Длина строки сохраняется из регистра eax после системного вызова read.

Модификация программы с in_out.asm

Задание: Модификация программы lab5-2.asm для вывода введенной строки обратно на экран с использованием подпрограмм из файла in_out.asm.

Выполнение:

- 1) Создана копия файла lab5-4.asm
- 2) Добавлен вызов подпрограммы sprintLF после ввода строки для ее вывода на экран
- 3) Программа собрана и протестирована

```
ssgaliev@linux ~/Загрузки % nasm -f elf32 lab5-4.asm
ssgaliev@linux ~/Загрузки % ld -m elf_i386 -o lab5-4 lab5-4.o
ssgaliev@linux ~/Загрузки % ./lab5-4
Введите строку: Галиев Самир Салаватович
Галиев Самир Салаватович

ssgaliev@linux ~/Загрузки %
```

Рисунок 7: Работа программы lab5-4 с использованием подпрограмм

Выводы

В ходе выполнения лабораторной работы были достигнуты следующие результаты:

1. Освоена работа с МС(Midnight Commander).
2. Изучена структура программ на NASM.
3. Освоены инструкции `mov` и `int`.
4. Реализован диалог с пользователем.
5. Освоена работа с внешними файлами.
6. Выполнены задания для самостоятельной работы.

Цель работы: приобретение практических навыков работы в Midnight Commander и освоены инструкций языка ассемблера `mov` и `int` — достигнута полностью.

Список литературы:

1. Midnight Commander Development Center. — 2021. — URL: <https://midnight-commander.org/>
2. NASM Assembly Language Tutorials. — 2021. — URL: <https://asmtutor.com/>
3. The NASM documentation. — 2021. — URL: <https://www.nasm.us/docs.php>
- 4.Расширенный ассемблер: NASM. — 2021. — URL: <https://www.opennet.ru/docs/RUS/nasm/>
5. Столяров А. Программирование на языке ассемблера NASM для ОС Unix. — 2-е изд. — М. : МАКС Пресс, 2011. — URL: http://www.stolyarov.info/books/asm_unix