

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра прикладной информатики и теории вероятностей

ОТЧЕТ

ПО ЛАБОРАТОРНОЙ РАБОТЕ № 5

дисциплина: Архитектура компьютера

Студент: Терентьев Максим Сергеевич

Группа: НКАбд-05-25

МОСКВА

2025 г.

Содержание

1. Цель работы.....	3
2. Порядок выполнения лабораторной работы.....	4
2.1. Подключение внешнего файла in_out.asm.....	5
3. Задание для самостоятельной работы.....	7
4. Вывод.....	9

1. Цель работы

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

2. Порядок выполнения лабораторной работы

Откроем Midnight Commander, перейдем в каталог `~/work/arch-pc`, создадим папку `lab05` и в этой папке создадим файл `lab5-1.asm`:

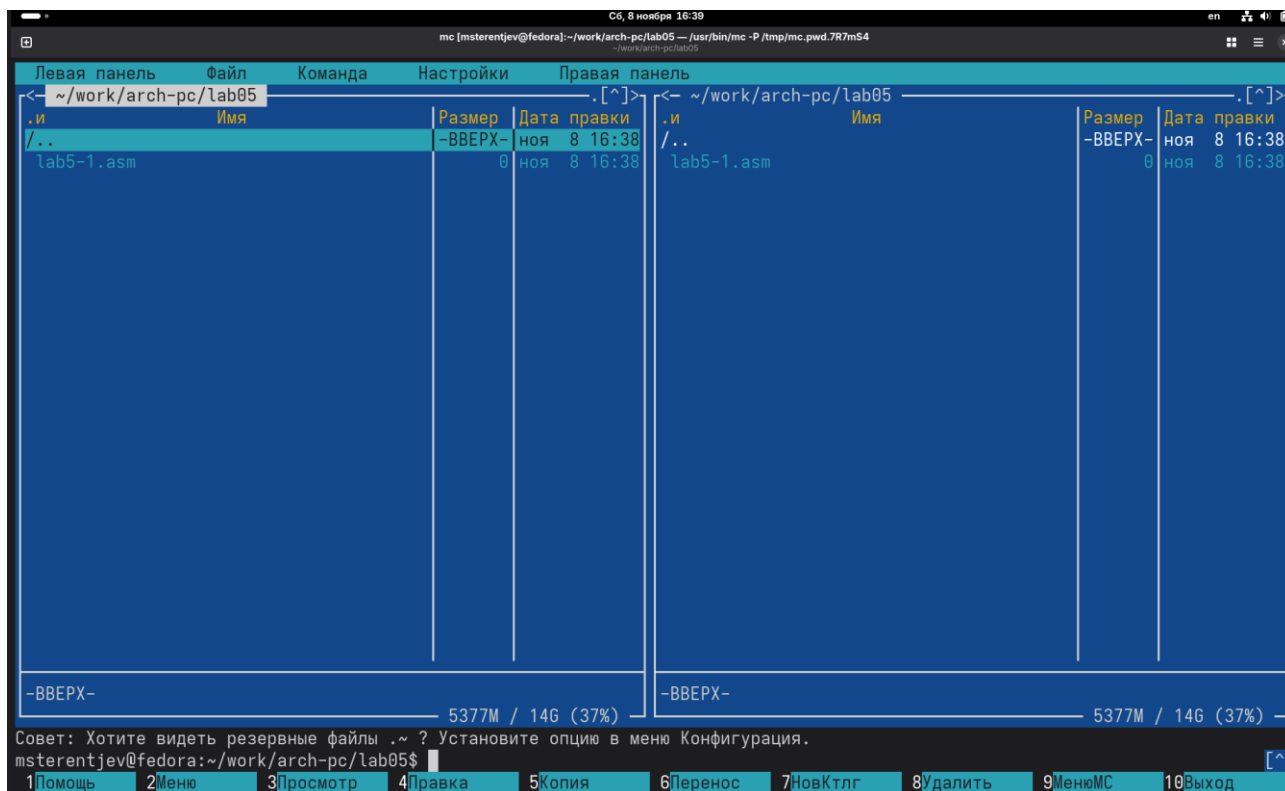


Рис. 1. Создание `lab5-1.asm` в midnight commander

С помощью F4 откроем файл `lab5-1.asm`, введем туда код программы из листинга 5.1:

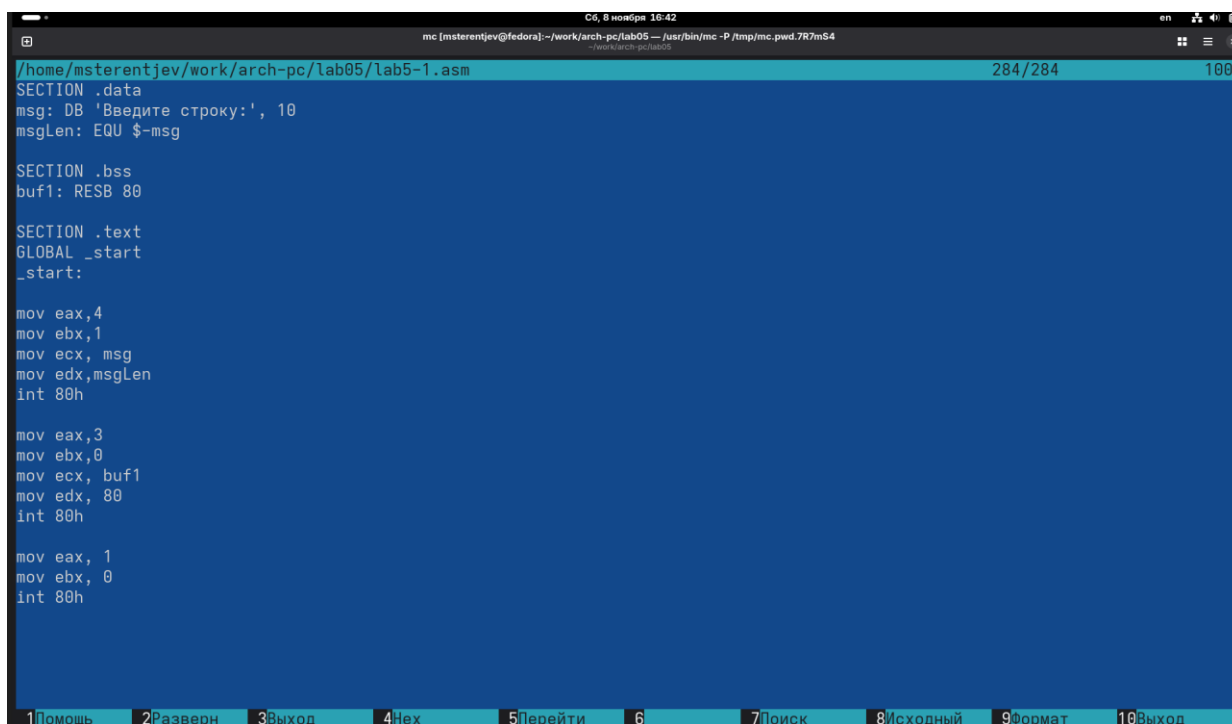


Рис. 2. Код программы в mcedit

Оттранслируем текст программы lab5-1.asm в объектный файл. Выполним компоновку объектного файла и запустим получим исполняемый файл. Запустим, введем ФИО:

```
msterentjev@fedora:~/work/arch-pc/lab05$ nasm -f elf lab5-1.asm
msterentjev@fedora:~/work/arch-pc/lab05$ ld -m elf_i386 -o lab5-1 lab5-1.o
msterentjev@fedora:~/work/arch-pc/lab05$ ./lab5-1
Введите строку:
Терентьев Максим Сергеевич
msterentjev@fedora:~/work/arch-pc/lab05$
```

Рис. 3. Компиляция и запуск lab5-1.asm

2.1. Подключение внешнего файла in_out.asm

Скачаем с ТУИС файл in_out.asm и скопируем файл в каталог с программой с помощью midnight commander:

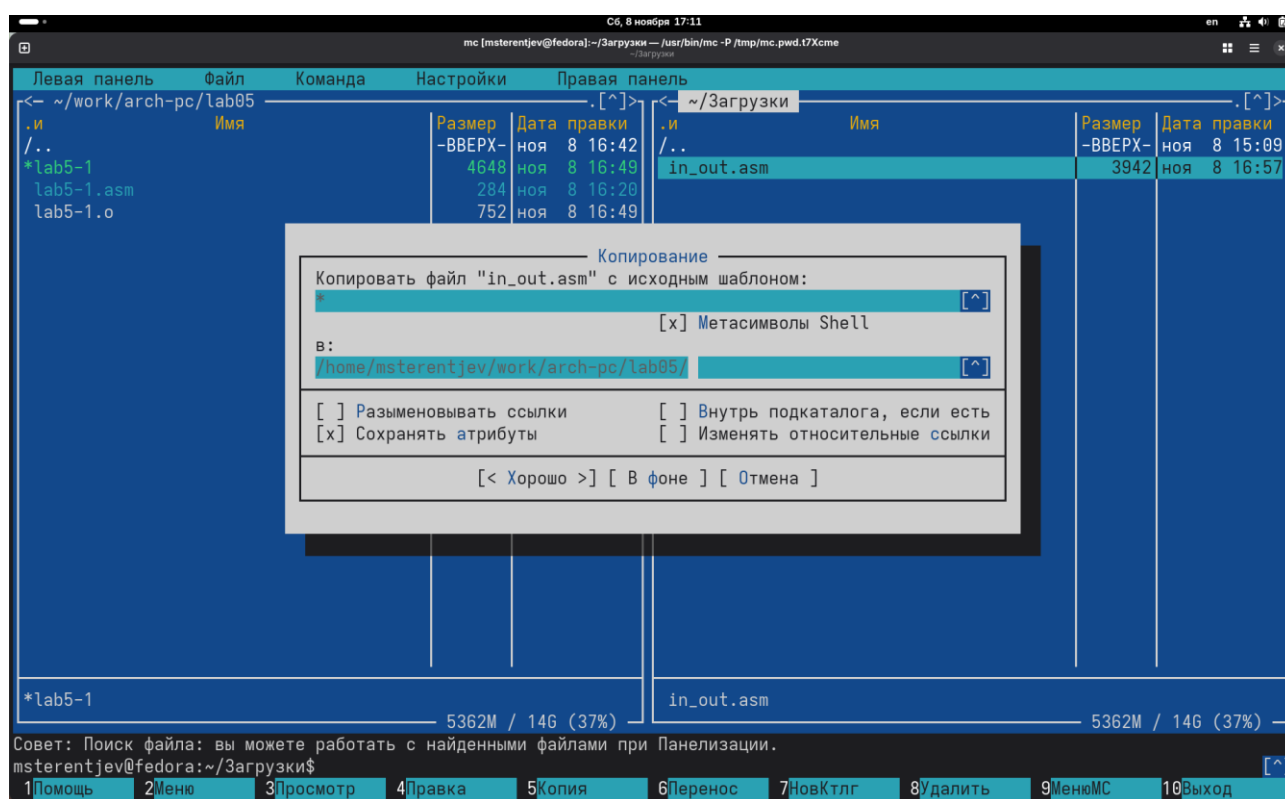


Рис. 4. Копирование in_out.asm в lab05

С помощью F5 создадим копию lab5-1.asm с именем lab5-2.asm:

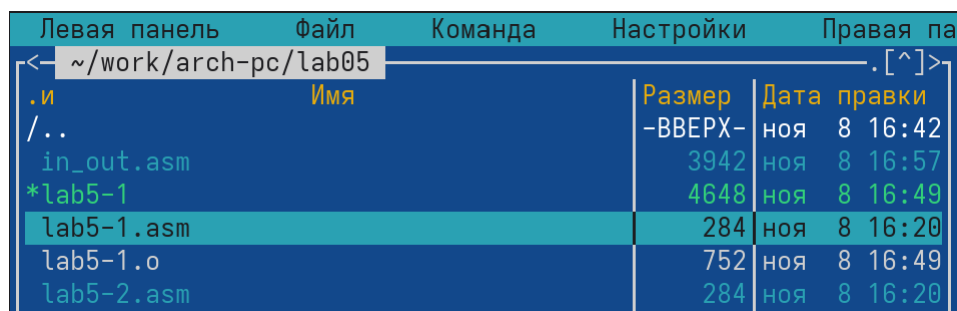


Рис. 5. Копия lab5-1.asm с именем lab5-2.asm

Исправим текст программы в файле lab5-2.asm с использованием подпрограмм из внешнего файла in_out.asm (подпрограммы sprintLF, sread и quit) в соответствии с листингом 5.2. Создадим исполняемый файл и проверим его работу:

```
%include 'in_out.asm' ; подключение внешнего файла

SECTION .data ; Секция инициированных данных
msg: DB 'Введите строку: ',0h ; сообщение

SECTION .bss ; Секция не инициированных данных
buf1: RESB 80 ; Буфер размером 80 байт

SECTION .text ; Код программы
GLOBAL _start ; Начало программы
_start: ; Точка входа в программу

mov eax, msg
call sprintLF ; вызов подпрограммы печати сообщения

mov ecx, buf1
mov edx, 80
call sread ; вызов подпрограммы ввода сообщения

call quit ; вызов подпрограммы завершения
```

Рис. 6. Измененный код lab5-2.asm

```
msterentjev@fedora:~/work/arch-pc/lab05$ nasm -f elf lab5-2.asm
msterentjev@fedora:~/work/arch-pc/lab05$ ld -m elf_i386 -o lab5-2 lab5-2.o
msterentjev@fedora:~/work/arch-pc/lab05$ ./lab5-2
Введите строку:
ssssssssssssss
msterentjev@fedora:~/work/arch-pc/lab05$
```

Рис. 7. Компиляция и проверка работы файла lab5-2

В файле lab5-2.asm заменим подпрограмму sprintLF на sprint. Создадим исполняемый файл и проверим его работу:

```
1 %include 'in_out.asm' ; подключение внешнего файла
2
3 SECTION .data ; Секция инициированных данных
4 msg: DB 'Введите строку: ',0h ; сообщение
5
6 SECTION .bss ; Секция не инициированных данных
7 buf1: RESB 80 ; Буфер размером 80 байт
8
9 SECTION .text ; Код программы
9 GLOBAL _start ; Начало программы
1 _start: ; Точка входа в программу
2
3 mov eax, msg ;
4 call sprint ; вызов подпрограммы печати сообщения
5
6 mov ecx, buf1 ;
7 mov edx, 80 ;
8 call sread ; вызов подпрограммы ввода сообщения
9
9 call quit ; вызов подпрограммы завершения
```

Рис. 8. Замена sprintLF на sprint

```
msterentjev@fedora:~/work/arch-pc/lab05$ gedit lab5-2.asm
msterentjev@fedora:~/work/arch-pc/lab05$ nasm -f elf lab5-2.asm
msterentjev@fedora:~/work/arch-pc/lab05$ ld -m elf_i386 -o lab5-2 lab5-2.o
msterentjev@fedora:~/work/arch-pc/lab05$ ls
in_out.asm lab5-1 lab5-1.asm lab5-1.o lab5-2 lab5-2.asm lab5-2.o
msterentjev@fedora:~/work/arch-pc/lab05$ ./lab5-2
Введите строку: amogus
msterentjev@fedora:~/work/arch-pc/lab05$
```

Рис. 9. Работа обновленного файла lab5-2

Можем заметить, что в отличие от прошлого файла и lab5-1, запрашиваемая строка выводится сразу же, а не на следующей строчке.

3. Задание для самостоятельной работы

- 1) Создадим копию файла lab5-1.asm. Внесем изменения в программу (без использования внешнего файла in_out.asm), так чтобы она работала по следующему алгоритму:

- вывести приглашение типа “Введите строку:”;
- ввести строку с клавиатуры;
- вывести введенную строку на экран.

И проверим его работу, введя ФИО:

Исправленный текст программы:

```
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
mov ebx, 1
mov ecx, msg
mov edx, msgLen
int 80h
```

```
; Ввод строки
mov eax, 3
mov ebx, 0
mov ecx, buf1
mov edx, 80
int 80h
```

```
; Сохраняем длину введенной строки
mov esi, eax
```

```
; Вывод введенной строки
mov eax, 4
mov ebx, 1
mov ecx, buf1
mov edx, esi
int 80h
```

```

msterentjev@fedora:~/work/arch-pc/lab05$ nasm -f elf lab5-1-1.asm
msterentjev@fedora:~/work/arch-pc/lab05$ ld -m elf_i386 -o lab5-1-1 lab5-1-1.o
msterentjev@fedora:~/work/arch-pc/lab05$ ./lab5-1-1
Введите строку:
Терентьев Максим Сергеевич
Терентьев Максим Сергеевич
msterentjev@fedora:~/work/arch-pc/lab05$

```

Рис. 10. Компиляция и работа программы lab5-1-1

- 2) Создадим копию файла lab5-2.asm. Исправим текст программы с использованием подпрограмм из внешнего файла in_out.asm, так чтобы она работала по следующему алгоритму:

- вывести приглашение типа “Введите строку:”;
- ввести строку с клавиатуры;
- вывести введенную строку на экран.

И проверим ее работу, введя ФИО:

Исправленный текст программы:

```
%include 'in_out.asm'      ; подключение внешнего файла
```

```
SECTION .data              ; Секция инициированных данных
msg: DB 'Введите строку: ',0h ; сообщение
```

```
SECTION .bss               ; Секция не инициированных данных
buf1: RESB 80              ; Буфер размером 80 байт
```

```
SECTION .text              ; Код программы
GLOBAL _start              ; Начало программы
_start:                    ; Точка входа в программу
; Вывод приглашения
mov eax, msg
call sprint
```

```
; Ввод строки
mov ecx, buf1
mov edx, 80
call read
```

```
; Вывод введенной строки
mov eax, buf1
call sprintLF
```

```
; Завершение программы
call quit
```

```

msterentjev@fedora:~/work/arch-pc/lab05$ gedit lab5-2-1.asm
msterentjev@fedora:~/work/arch-pc/lab05$ nasm -f elf lab5-2-1.asm
msterentjev@fedora:~/work/arch-pc/lab05$ ld -m elf_i386 -o lab5-2-1 lab5-2-1.o
msterentjev@fedora:~/work/arch-pc/lab05$ ./lab5-2-1
Введите строку: amogus
amogus

msterentjev@fedora:~/work/arch-pc/lab05$ █

```

Рис. 11. Выполнение lab5-2-1

4. Вывод

Освоен процесс компиляции и сборки программ на ассемблере NASM в Linux. Практически изучены два подхода программирования: с использованием системных вызовов и с внешними библиотеками. Созданы рабочие программы, выполняющие ввод и вывод строк. Успешно применены команды компиляции (`nasm`) и линковки (`ld`) для генерации исполняемых файлов. Получены навыки работы со структурой ассемблерной программы и основными секциями данных!!!