

# **Отчет по лабораторной работе №5**

**Основы работы с Midnight Commander**

Ромицына Анастасия Романовна

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Задание</b>	<b>6</b>
<b>3</b>	<b>Выполнение лабораторной работы</b>	<b>7</b>
3.1	Задание для самостоятельной работы . . . . .	15
<b>4</b>	<b>Выводы</b>	<b>19</b>

## Список иллюстраций

3.1	Вводим команду <code>mc</code> . . . . .	7
3.2	Переходим в каталог . . . . .	8
3.3	Создаем каталог функциональной клавишей F7 . . . . .	8
3.4	Воспользуемся командой <code>touch</code> . . . . .	9
3.5	Открываем файл функциональной клавишей, заполняем и сохраняем . . . . .	9
3.6	Открываем файл и убеждаемся, что файл содержит текст программы . . . . .	10
3.7	Проверяем, как работает данная программа . . . . .	10
3.8	Скачиваем файл . . . . .	11
3.9	Копируем скаченный файл . . . . .	11
3.10	Создаем копию файла клавишей F6 . . . . .	12
3.11	Проверяем скопировался ли файл . . . . .	12
3.12	Открываем и заполняем файл . . . . .	13
3.13	Смотрим, как сработала программа . . . . .	13
3.14	Редактируем файл . . . . .	14
3.15	Смотрим, как сработала программа и сравниваем с прошлой . . . . .	14
3.16	Создаем копию файла <code>lab5-1.asm</code> . . . . .	15
3.17	Редактируем файл . . . . .	16
3.18	Проверяем правильность написания программы . . . . .	16
3.19	Создаем копию файла <code>lab5-2.asm</code> . . . . .	17
3.20	Редактируем файл . . . . .	17
3.21	Проверяем правильность написания программы . . . . .	18

## Список таблиц

# 1 Цель работы

Освоить инструкции языка ассамблера mov. Преобрести знания использования Midnight Commander.

## 2 Задание

Написать 2 программы по примеру и впоследствии изменить их по условию.

### 3 Выполнение лабораторной работы

Открываем Midnight Commander (рис. 3.1).

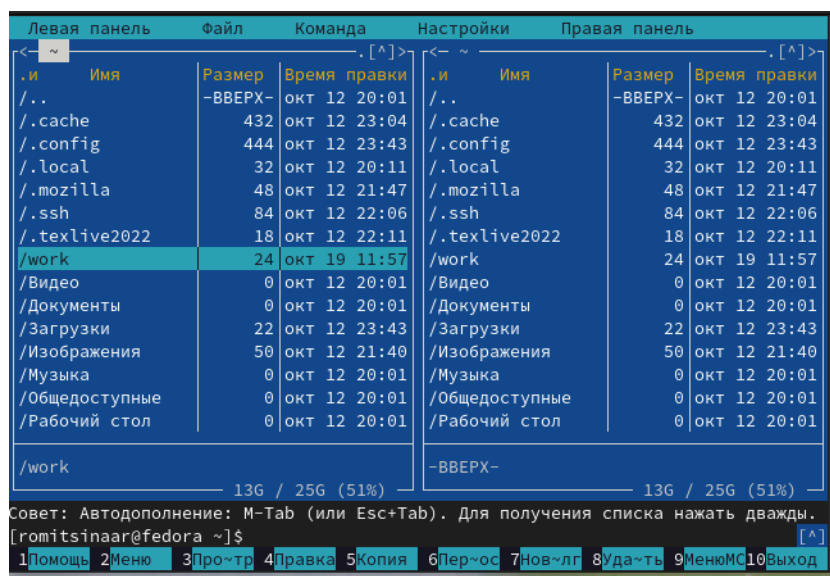


Рис. 3.1: Вводим команду mc

Переходим в каталог, созданный при выполнении 4 ЛБ (рис. 3.2).

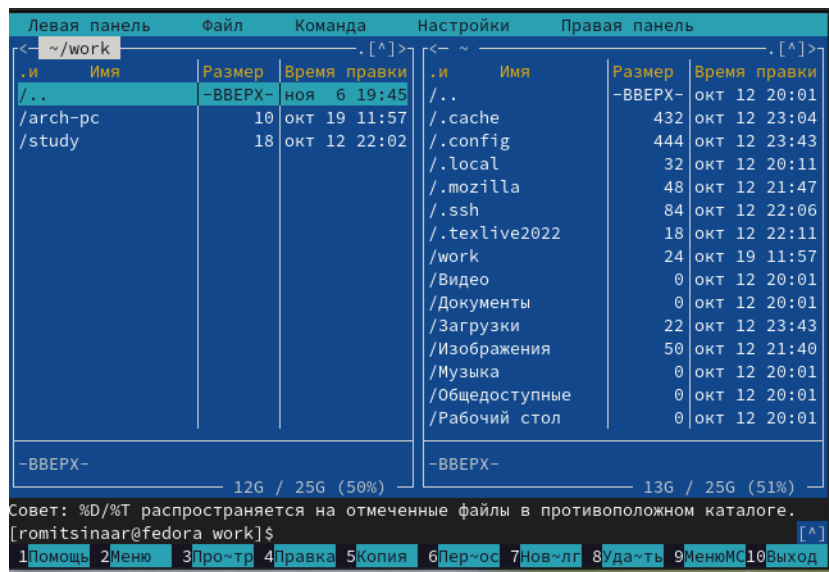


Рис. 3.2: Переходим в каталог

Создаем каталог lab05 (рис. 3.3).

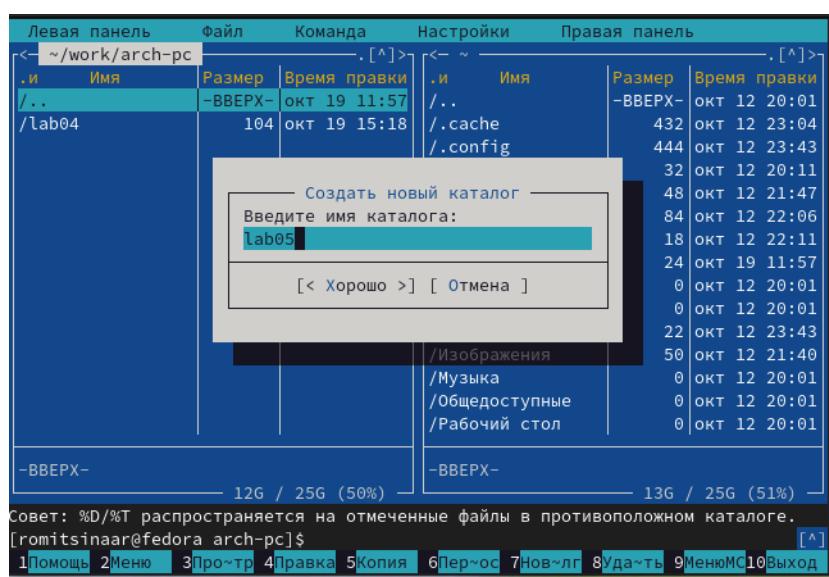


Рис. 3.3: Создаем каталог функциональной клавишей F7

Создаем файл lab5-1.asm (рис. 3.4).



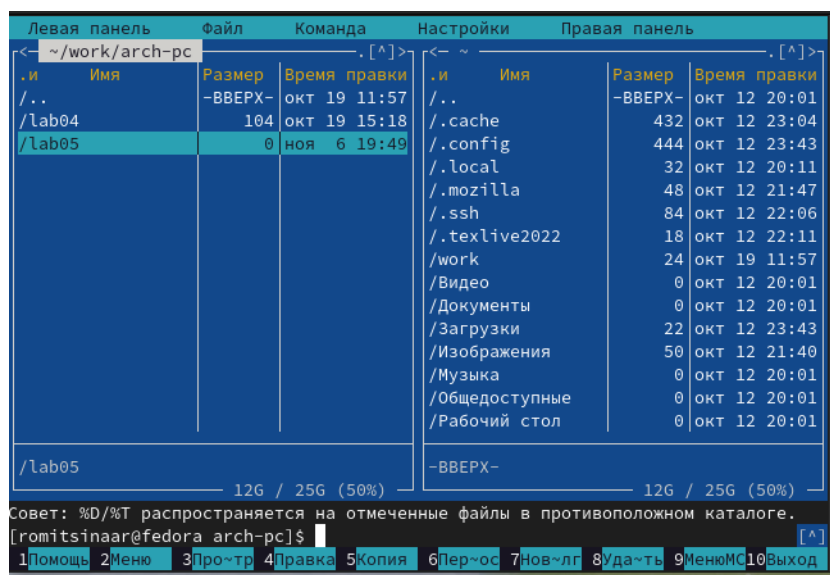


Рис. 3.4: Воспользуемся командой touch

Открываем файл для редактирования и заполняем его по листингу (рис. 3.5).

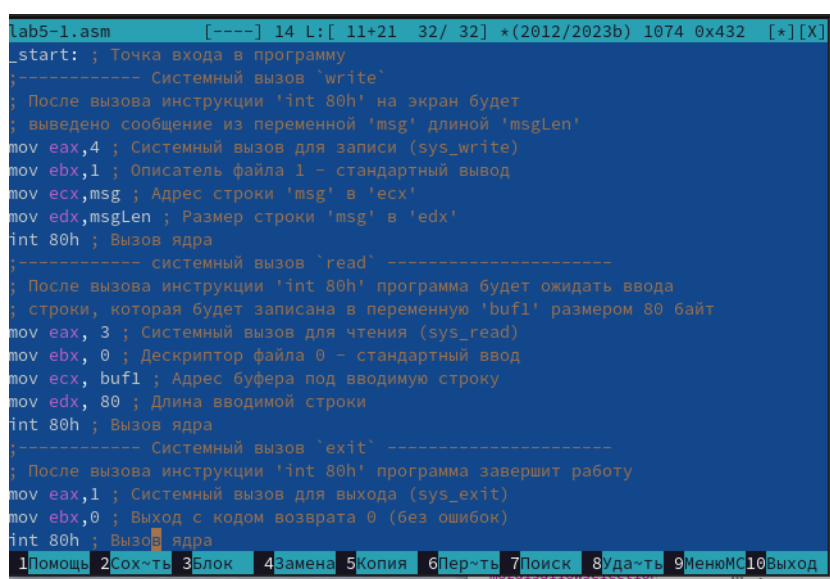


Рис. 3.5: Открываем файл функциональной клавишей, заполняем и сохраняем

Открываем файл для просмотра (рис. 3.6).

```

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

```

Рис. 3.6: Открываем файл и убеждаемся, что файл содержит текст программы

Транслируем текст программы и запускаем исполняемый файл (рис. 3.7).

```

[romitsinaar@fedora lab05]$ ld -m elf_i386 -o lab5-1 lab-1.o
ld: невозможно найти lab-1.o: Нет такого файла или каталога
[romitsinaar@fedora lab05]$ ld -m elf_i386 -o lab5-1 lab5-1.o
[romitsinaar@fedora lab05]$ ./lab5-1
Введите строку:
Ромицына Анастасия Романовна
[romitsinaar@fedora lab05]$

```

Рис. 3.7: Проверяем, как работает данная программа

Скачиваем файл со страницы курса (рис. 3.8).

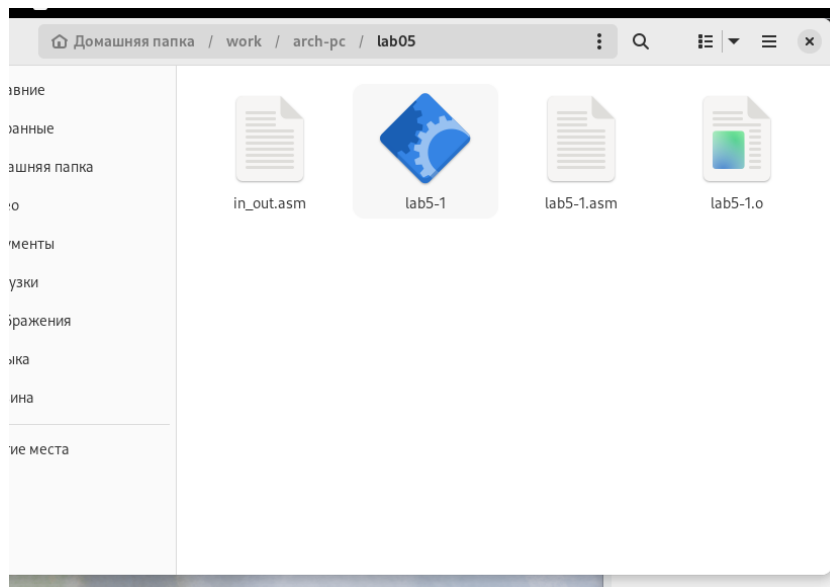


Рис. 3.8: Скачиваем файл

Копируем файл в нужную директорию (рис. 3.9).

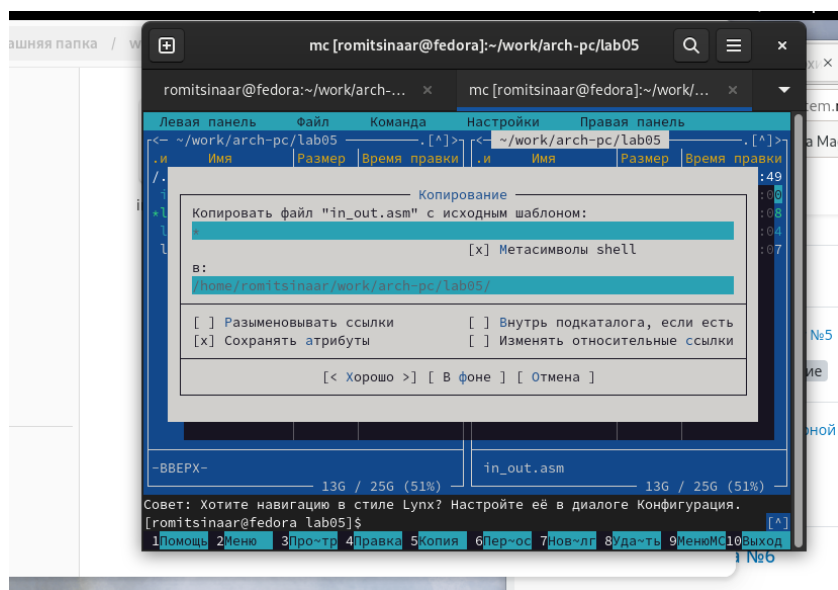


Рис. 3.9: Копируем скаченный файл

Создаем копию файла lab5-1.asm (рис. 3.10).

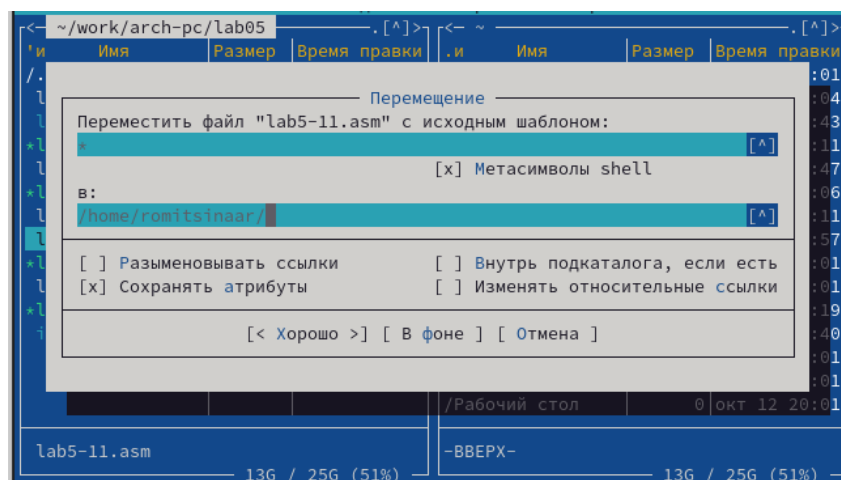


Рис. 3.10: Создаем копию файла клавишей F6

Проверяем созданный файл (рис. 3.11).

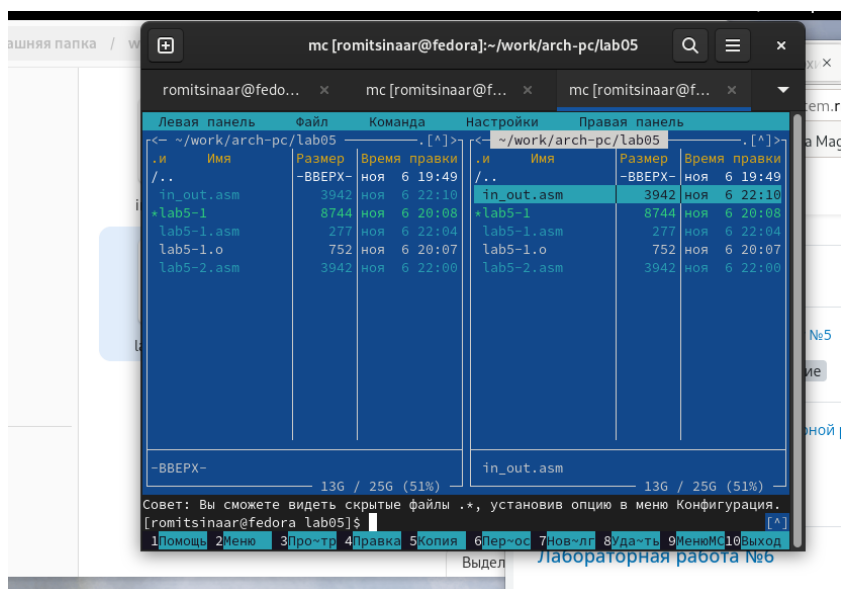


Рис. 3.11: Проверяем скопировался ли файл

Открываем новый файл и заполняем его в соответствии с листингом (рис. 3.12).

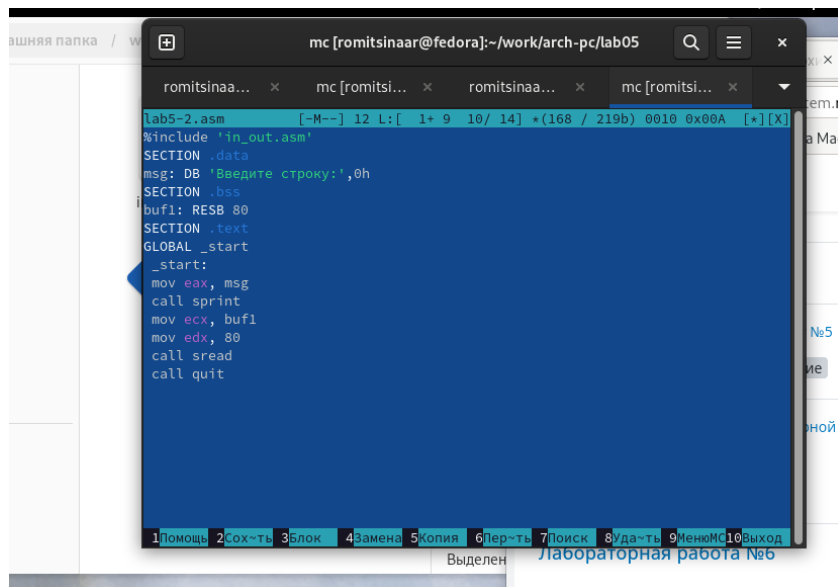


Рис. 3.12: Открываем и заполняем файл

Транслируем и запускаем новый файл (рис. 3.13).

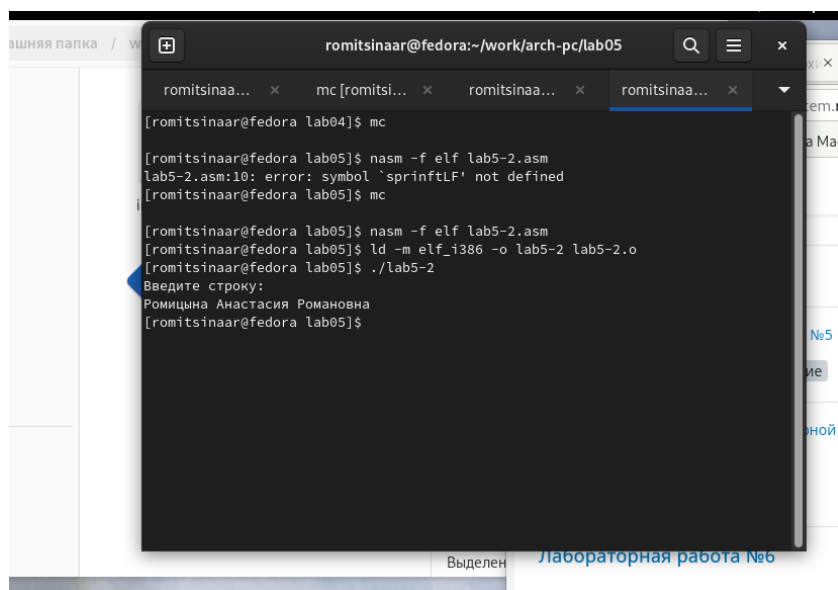
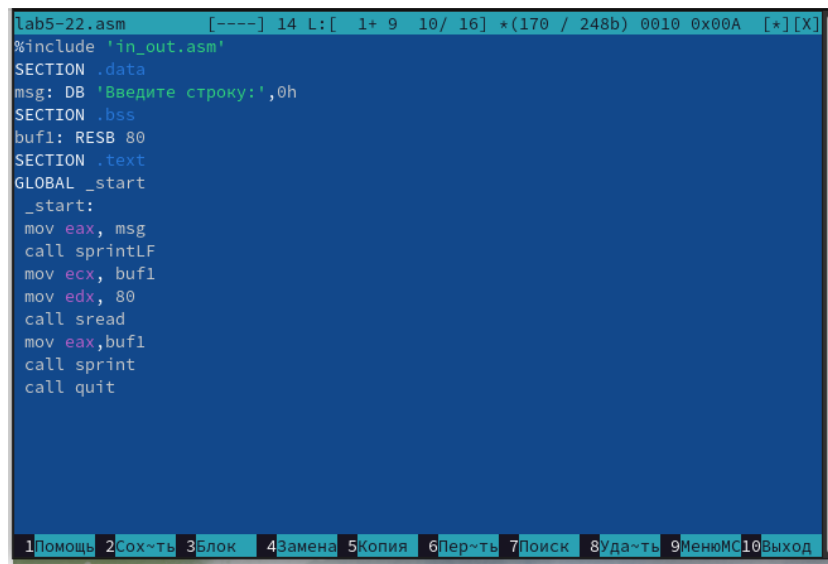


Рис. 3.13: Смотрим, как сработала программа

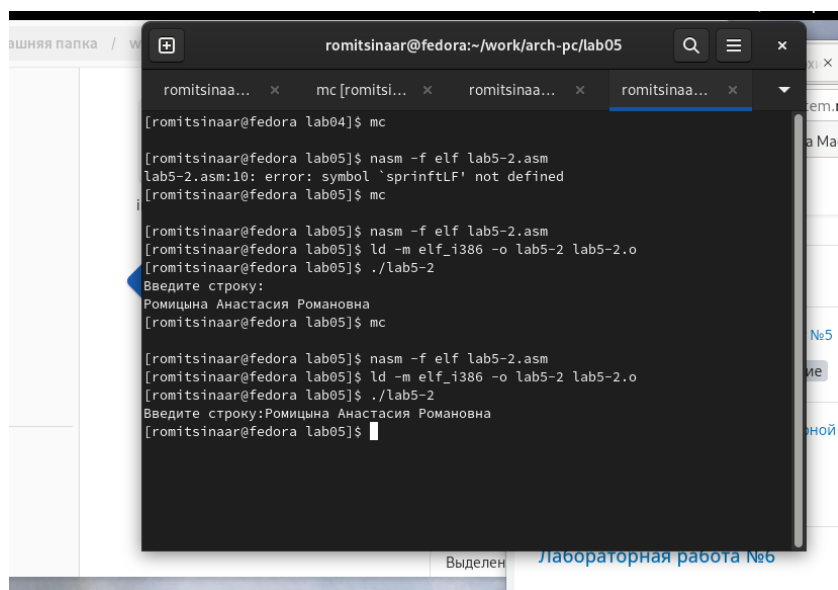
Снова открываем файл для редактирования и меняем `sprintLF` на `sprint` (рис. 3.14).



```
lab5-22.asm  [----] 14 L: [ 1+ 9 10/ 16] *(170 / 248b) 0010 0x00A [*][X]
#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 sprintf
mov ecx, buf1
mov edx, 80
call sread
mov eax, buf1
call sprint
call quit
```

Рис. 3.14: Редактируем файл

Транслируем и запускаем файл(рис. 3.15).



```
romitsinaa@fedora:~/work/arch-pc/lab05
romitsinaa... x mc[romitsi... x romitsinaa... x romitsinaa... x
[romitsinaa@fedora lab04]$ mc
[romitsinaa@fedora lab05]$ nasm -f elf lab5-2.asm
lab5-2.asm:10: error: symbol 'sprintf' not defined
[romitsinaa@fedora lab05]$ mc
[romitsinaa@fedora lab05]$ nasm -f elf lab5-2.asm
[romitsinaa@fedora lab05]$ ld -m elf_i386 -o lab5-2 lab5-2.o
[romitsinaa@fedora lab05]$ ./lab5-2
Введите строку:
Ромицына Анастасия Романовна
[romitsinaa@fedora lab05]$ mc
[romitsinaa@fedora lab05]$ nasm -f elf lab5-2.asm
[romitsinaa@fedora lab05]$ ld -m elf_i386 -o lab5-2 lab5-2.o
[romitsinaa@fedora lab05]$ ./lab5-2
Введите строку:Ромицына Анастасия Романовна
[romitsinaa@fedora lab05]$
```

Рис. 3.15: Смотрим, как сработала программа и сравниваем с прошлой

Таким образом можем понять, что команда `sprint` выводит текст в той же строке, а `sprintf` переносит на новую строку.

### 3.1 Задание для самостоятельной работы

Создаем копию файла lab5-1.asm (рис. 3.16).

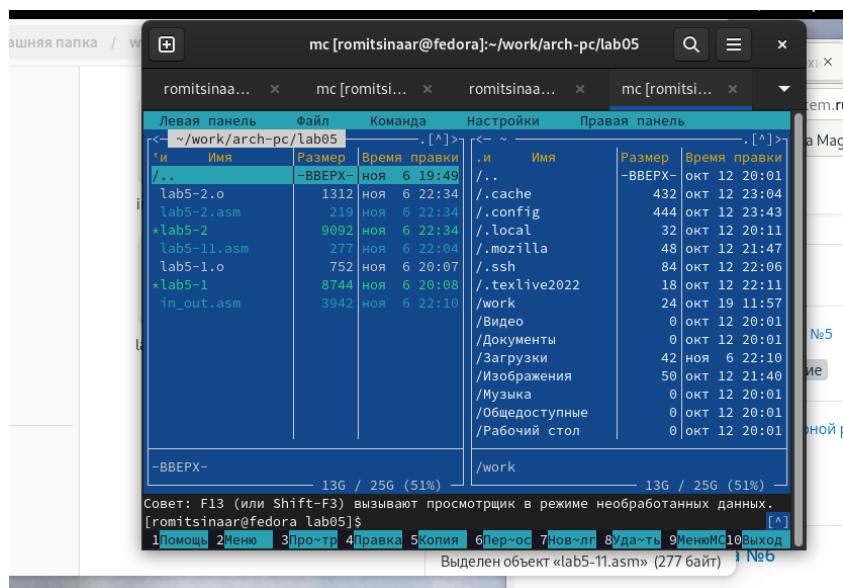


Рис. 3.16: Создаем копию файла lab5-1.asm

Редактируем файл, чтобы введенный текст с клавиатуры выводился в консоль (рис. 3.17).

```
lab5-11.asm  [----]  1  L: [ 11+14  25/ 32] *(295 / 333b) 0110 0x06E [*][X]
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,80
int 80h

mov eax,1
mov ebx,0
int 80h
```

1Помощь 2Сох~ть 3Блок 4Замена 5Копия 6Пер~ть 7Поиск 8Уда~ть 9МенюМС10Выход

Рис. 3.17: Редактируем файл

Транслируем файл и запускаем программу (рис. 3.18).

```
lab5-1.asm:6: error: label or instruction expected at start of line
lab5-1.asm:7: error: parser: instruction expected

[romitsinaar@fedora lab05]$ ld -m elf_i384 -o lab5-1 lab5-1.o
ld: не распознан режим эмуляции: elf_i384
Поддерживаемые эмуляции: elf_x86_64 elf32_x86_64 elf_i386 elf_iamcu i386pep i386
pe elf64bpf

[romitsinaar@fedora lab05]$ ./lab5-1
bash: ./lab5-1: Нет такого файла или каталога

[romitsinaar@fedora lab05]$ nasm -f elf lab5-1.asm
lab5-1.asm:6: error: label or instruction expected at start of line
lab5-1.asm:7: error: parser: instruction expected
[romitsinaar@fedora lab05]$ mc

[romitsinaar@fedora lab05]$ nasm -f elf lab5-1.asm
[romitsinaar@fedora lab05]$ ld -m elf_i386 -o lab5-1 lab-1.o
ld: невозможно найти lab-1.o: Нет такого файла или каталога
[romitsinaar@fedora lab05]$ ld -m elf_i386 -o lab5-1 lab5-1.o
[romitsinaar@fedora lab05]$ ./lab5-1
Введите строку:
Ромицына Анастасия Романовна
[romitsinaar@fedora lab05]$
```

Рис. 3.18: Проверяем правильность написания программы

Создаем копию файла lab5-2.asm (рис. 3.19).



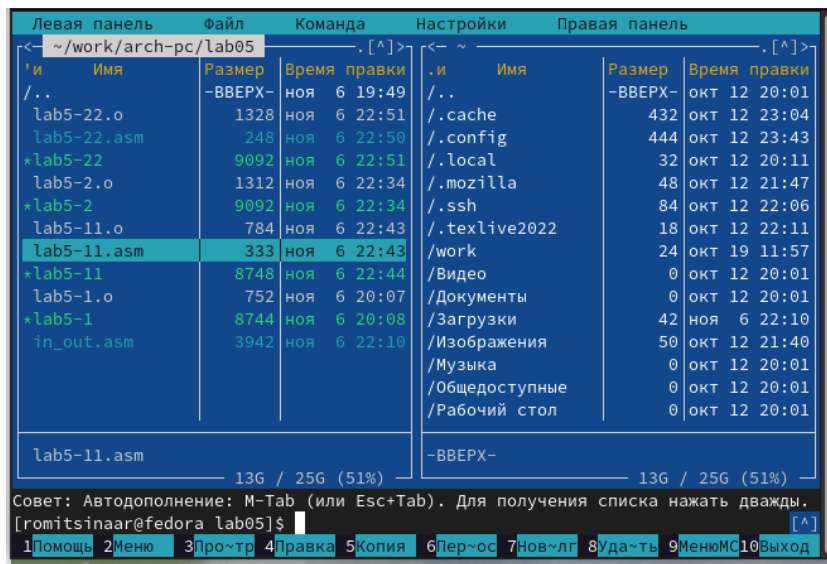


Рис. 3.19: Создаем копию файла lab5-2.asm

Редактируем файл, чтобы введенный текст с клавиатуры выводился в консоль (рис. 3.20).

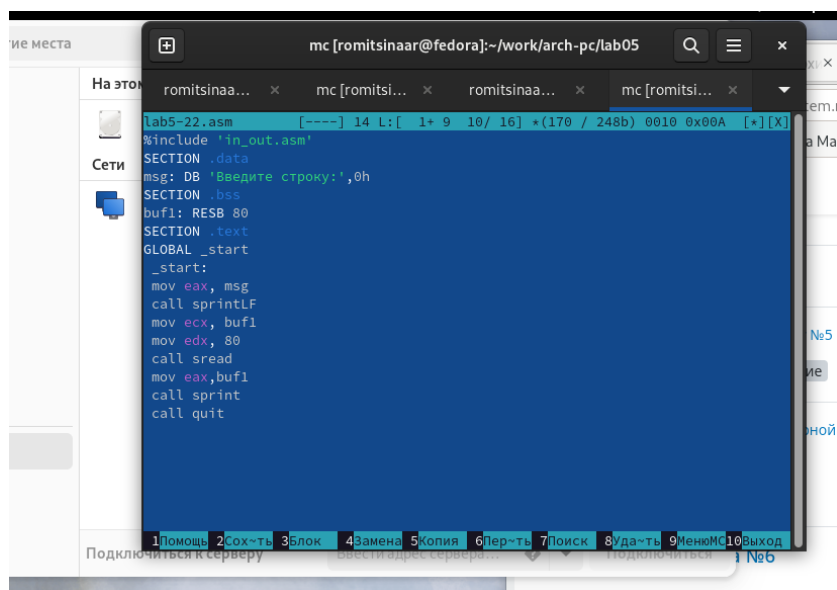


Рис. 3.20: Редактируем файл

Транслируем файл и запускаем программу (рис. 3.21).

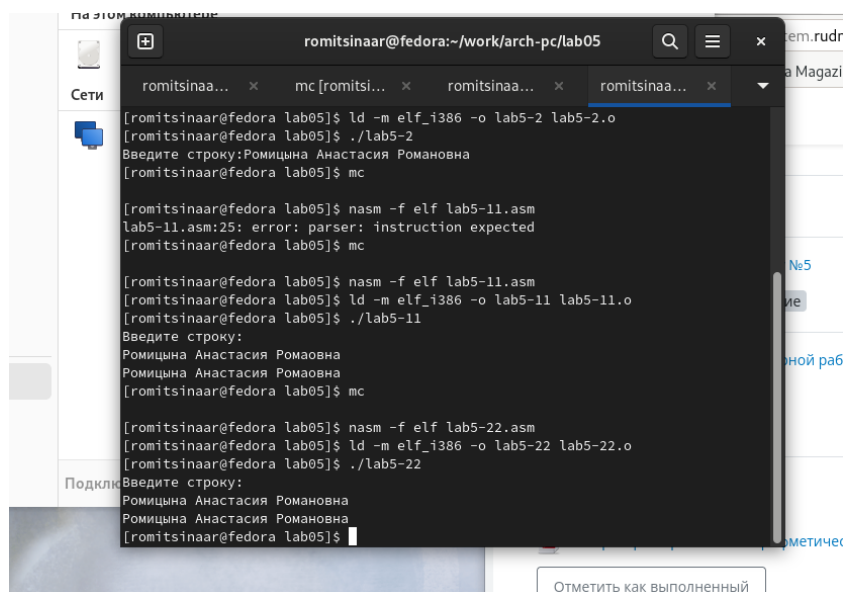


Рис. 3.21: Проверяем правильность написания программы

## 4 Выводы

Мы приобрели навыки работы с Midnight Commander и освоили инструкции  
mov