

# **Лабораторная работа №6**

**Основы работы с Midnight Commander. Структура программ на языке  
ассемблера NASM. Системные вызовы в ОС GNU Linux**

Турсунбоев Сардорбек

# Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение лабораторной работы	7
4	Выводы	19

## Список иллюстраций

3.1	Открытие Midnight Commander через терминал . . . . .	7
3.2	Переход в каталог arch-pc . . . . .	8
3.3	Создание каталога в Midnight Commander . . . . .	8
3.4	Переход в созданный нами каталог . . . . .	9
3.5	Создание файла lab5-1.asm . . . . .	9
3.6	Проверка правильности создания файла . . . . .	10
3.7	Открытие файла в редакторе mcedit . . . . .	11
3.8	Код, написанный по образцу . . . . .	11
3.9	Файл, открытый для просмотра . . . . .	12
3.10	Трансляция, компоновка и запуск lab5-1.asm . . . . .	12
3.11	Копирование файла в Midnight Commander . . . . .	13
3.12	Файл lab5-2.asm в рабочем каталоге . . . . .	14
3.13	Код, написанный по образцу с использованием подпрограмм из внешнего файла in_out.asm . . . . .	15
3.14	Работа файла lab5-2.asm . . . . .	15
3.15	Замена подпрограммы sprintLF на sprint . . . . .	16
3.16	Результат работы lab5-2.asm после замены sprintLF на sprint . . .	16
3.17	Код в файле lab5-1.asm . . . . .	17
3.18	Проверка работоспособности кода . . . . .	17
3.19	Написанный в lab5-2.asm код . . . . .	18
3.20	Проверка работоспособности lab5-2.asm . . . . .	18

## **Список таблиц**

# 1 Цель работы

Целью данной лабораторной работы является приобретение навыков работы с программой Midnight Commander и освоение инструкций `mov` и `int` в языке ассемблера NASM.

## 2 Задание

1. Основы работы в Midnight Commander
2. Структура программ на языке ассемблера NASM
3. Использование внешних файлов в языке ассемблера NASM
- 4.

### 3 Выполнение лабораторной работы

Открою Midnight Commander через терминал командой mc(Рис.1).

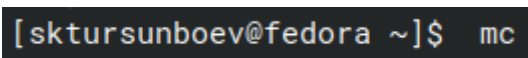
A terminal window with a dark background. The prompt is [sktursunboev@fedora ~]\$ and the command mc is entered.

Рис. 3.1: Открытие Midnight Commander через терминал

Перейду в созданный мной в ходе лабораторной работы №4 каталог arch-rc(Рис.2).

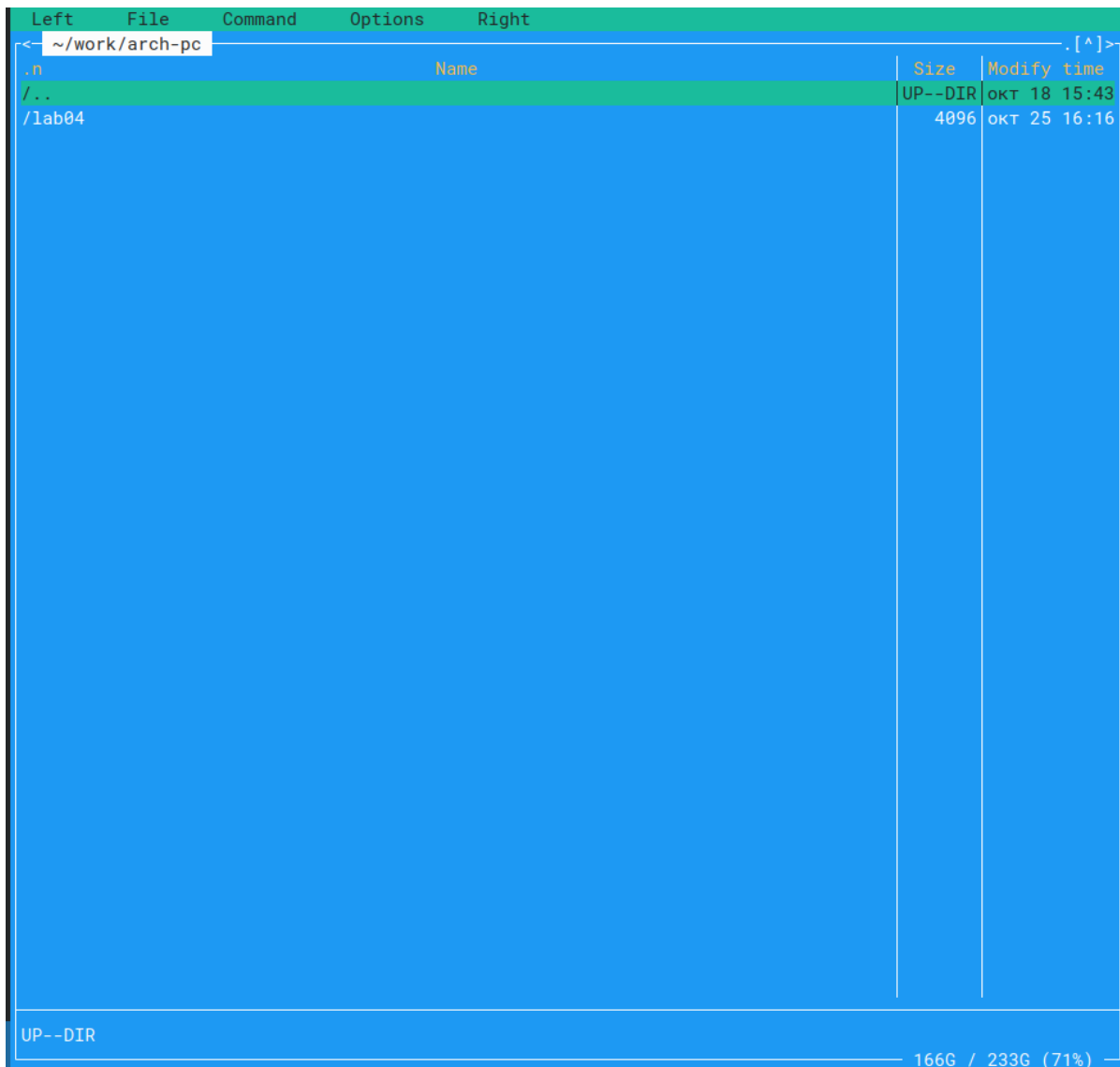


Рис. 3.2: Переход в каталог arch-rc

В Midnight Commander создам новый каталог lab05(Рис.3).

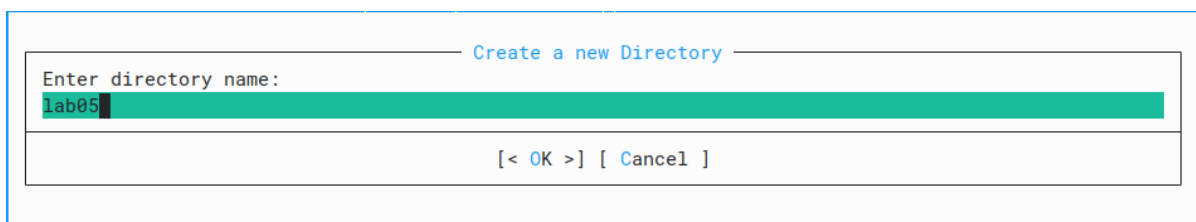


Рис. 3.3: Создание каталога в Midnight Commander



Перейду в созданный мной каталог(Рис.4).

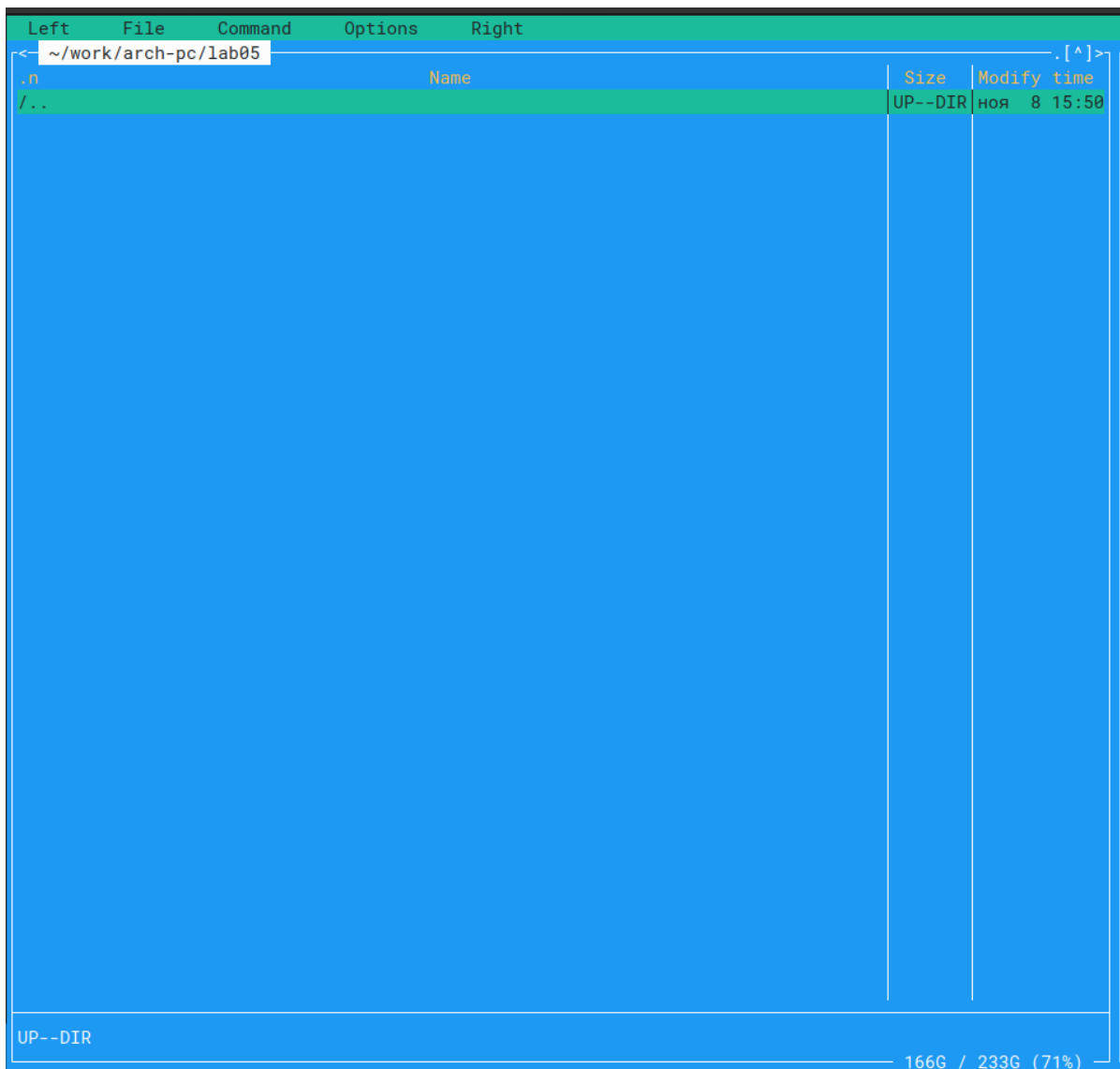


Рис. 3.4: Переход в созданный нами каталог

Утилитой touch создам файл lab5-1.asm(Рис.5).



Рис. 3.5: Создание файла lab5-1.asm

Убеждаюсь что файл действительно создан(Рис.6).

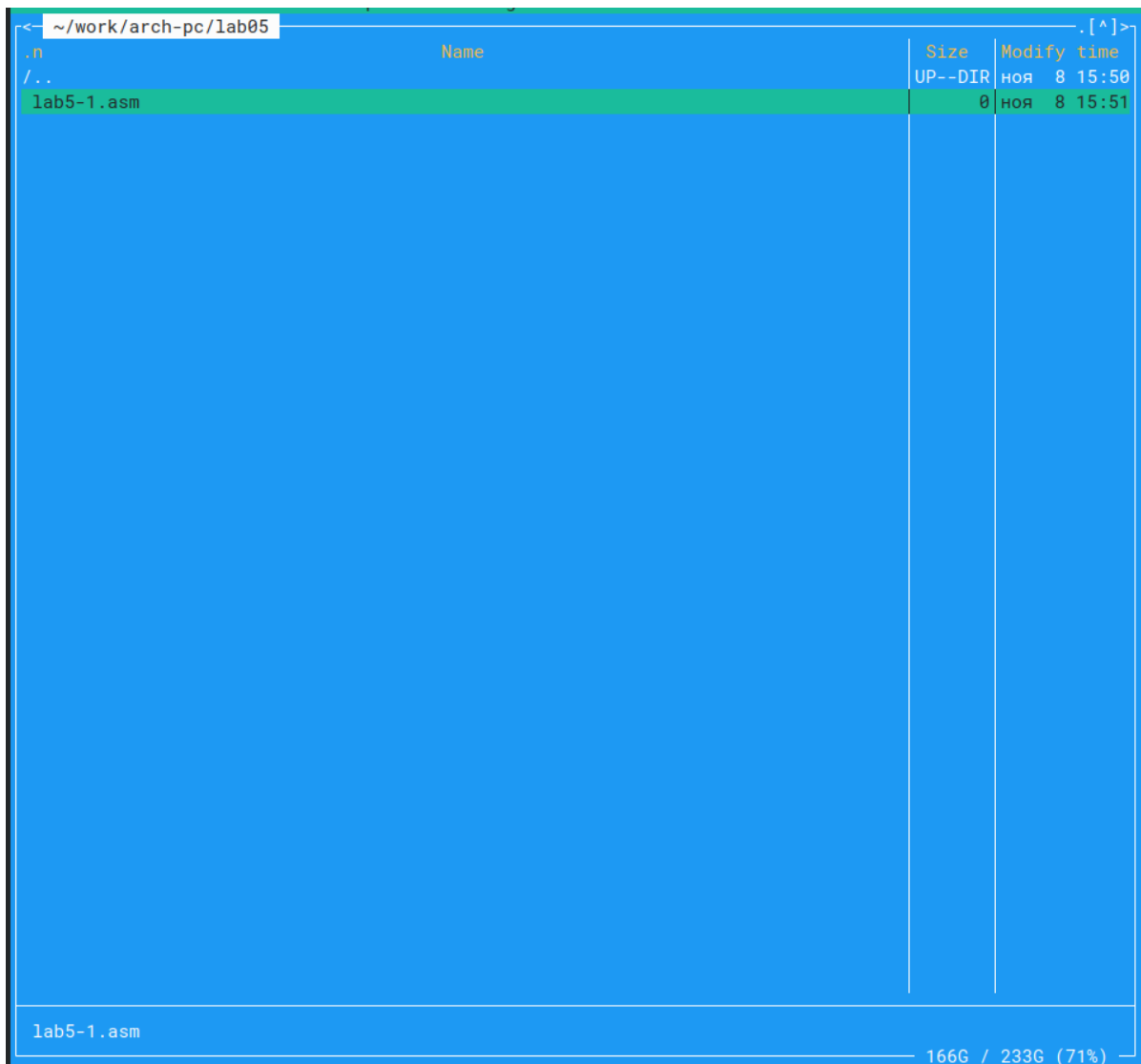


Рис. 3.6: Проверка правильности создания файла

Открою файл `lab5-1.asm` во встроенный редакторе `mcedit`(Рис.7).

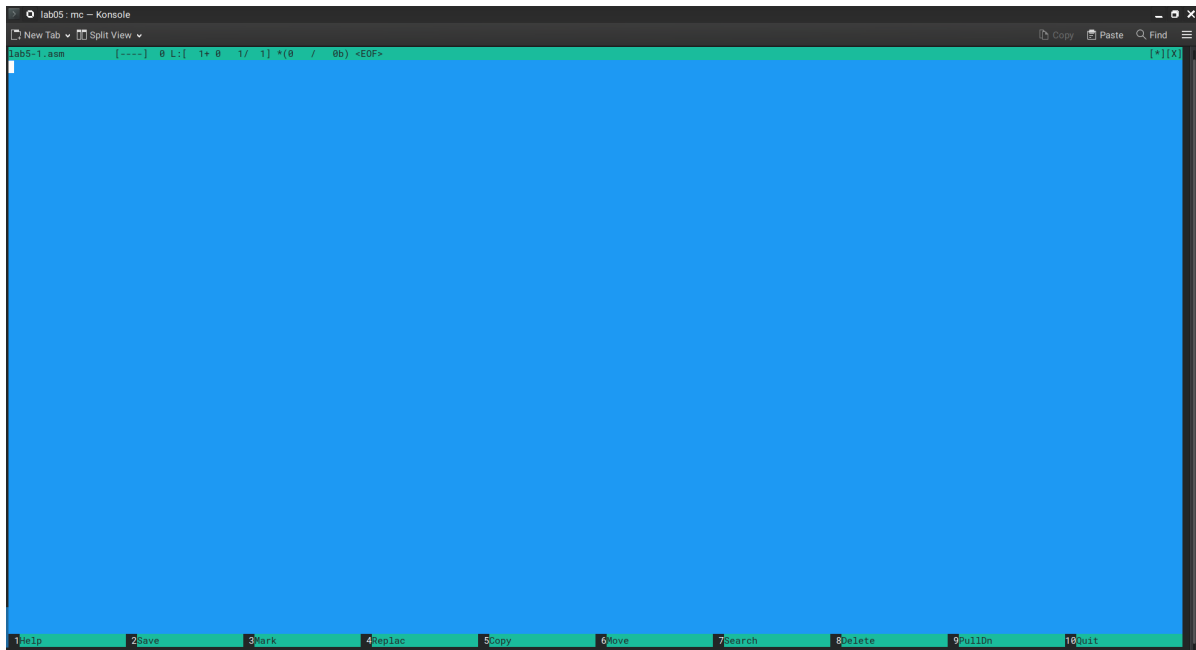


Рис. 3.7: Открытие файла в редакторе mcedit

Напишу код по образцу(Рис.8).

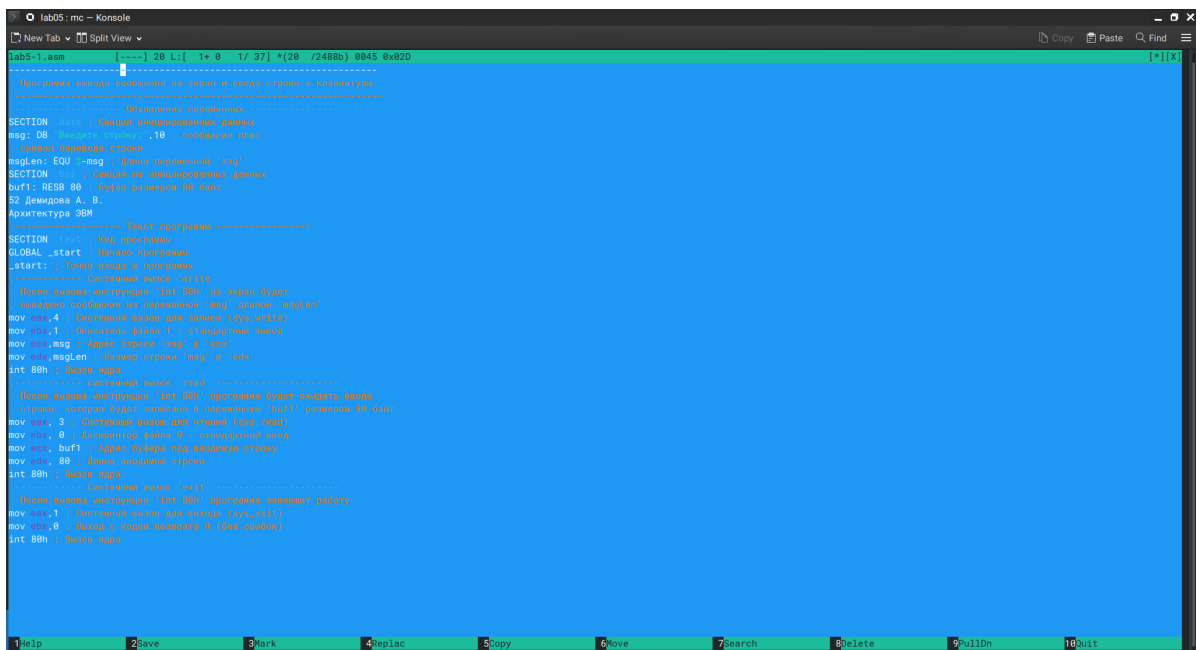


Рис. 3.8: Код, написанный по образцу

В Midnight Commander открою файл для просмотра и удостоверюсь что правильно написали код(Рис.9).

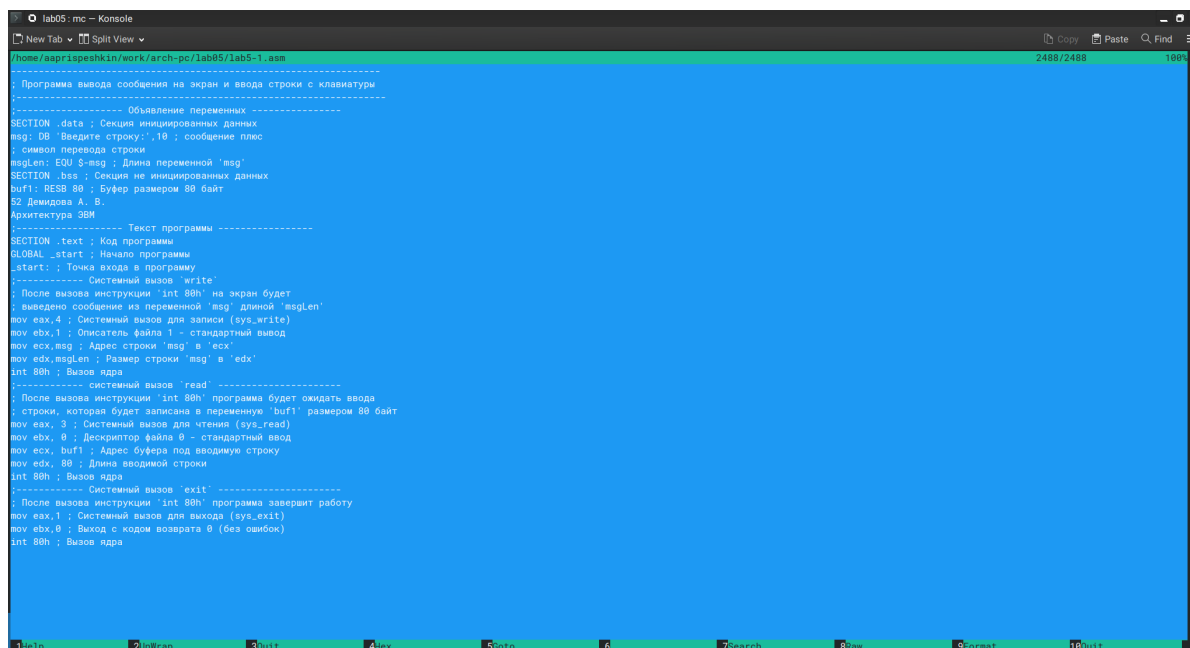


Рис. 3.9: Файл, открытый для просмотра

Оттранслирую и отправлю на компоновку lab5-1.asm, затем запущу полученный файл для проверки его работоспособности(Рис.10).

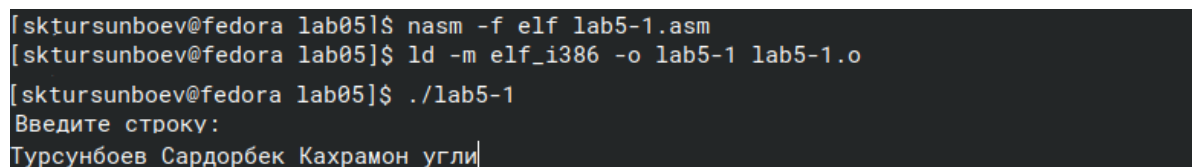


Рис. 3.10: Трансляция, компоновка и запуск lab5-1.asm

Скопирую файл lab5-1.asm в этот же каталог и назову его lab5-2.asm(Рис.11).

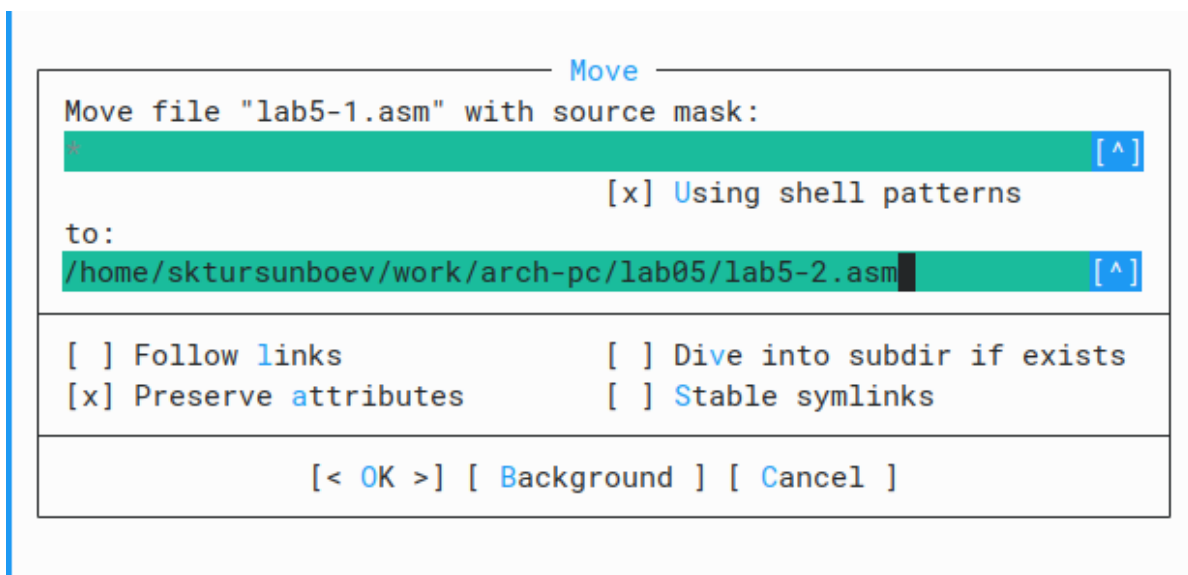


Рис. 3.11: Копирование файла в Midnight Commander

Убеждаюсь в правильности копирования файла(Рис.12).

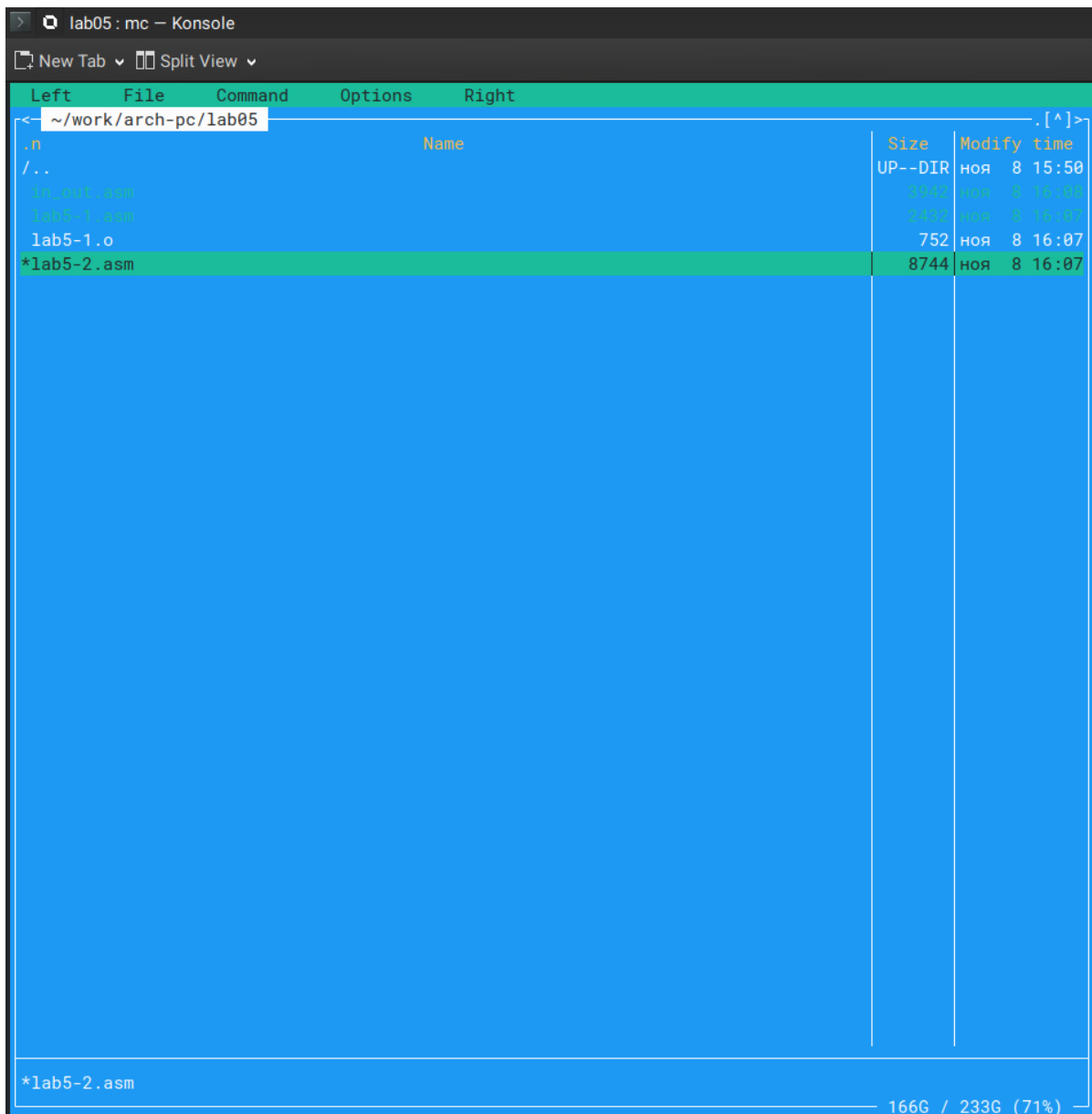


Рис. 3.12: Файл lab5-2.asm в рабочем каталоге

По образцу напишу код, включу в него внешний файл in\_out.asm и использую доступные подпрограммы(Рис.13).







```

lab5-1.asm [----] 3 L: [ 1+14 15/ 22] *(819 /13136) 0832 0x020 [X]
SECTION .text ; Секция исполняемых данных
msg: DB "Турсунбоев", 10 ; строение строки
msgLen: EQU $-msg ; длина переменной 'msg'
SECTION .bss ; Секция не-инициализированных данных
buf1: RESB 80 ; буфер размером 80 байт
SECTION .start ; Начало программы
GLOBAL _start ; Начало входа в программу
_start: ; Вход в код программы
mov eax, 4 ; Системный вызов для записи (sys_write)
mov ebx, 1 ; Файловый дескриптор 1 - стандартный вывод
mov ecx, msg ; Адрес строки 'msg' в 'ecx'
mov edx, msgLen ; Размер строки 'msg' в 'edx'
int 80h ; Вход в код
mov ebx, 3 ; Системный вызов для чтения (sys_read)
mov ecx, 0 ; Дескриптор файла 0 - стандартный ввод
mov edx, buf1 ; Адрес буфера под вводную строку
mov esi, 80 ; Длина ожидаемой строки
int 80h ; Вход в код
mov ebx, 1 ; Системный вызов для вывода (sys_exit)
mov ecx, 0 ; Выход с кодом возврата 0 (без ошибок)
int 80h ; Выход в код

```

Рис. 3.17: Код в файле lab5-1.asm

Проверю его работоспособность(Рис.18).

```

[sktursunboev@fedora lab05]$ ./lab5-1
Введите строку:
Турсунбоев

```

Рис. 3.18: Проверка работоспособности кода

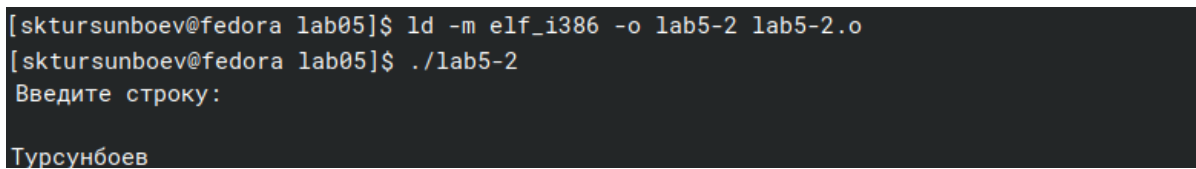
Скопирую файл lab5-2.asm и напишу там код для написания строки, запроса ввода от пользователя и вывода введённой строки на экран, в этот раз пользуясь подпрограммами из in\_out.asm(Рис.19)



```
lab5-2.asm [M--] 13 L: 1* 1 2/ 181 *(35 / 2676) 8010 0x00A [F][X]
#include <stdio.h>
SECTION .text
msg: DB "TurSunboev", 0h
SECTION .data
buf1: RESB 80
SECTION .bss
GLOBAL _start
_start:
mov eax, msg
call sprint
mov ecx, buf1
mov edx, 80
call sread
mov ebx, 4
mov ebx, 1
mov ecx, buf1
int 80h
call quit
```

Рис. 3.19: Написанный в lab5-2.asm код

Проверю работоспособность кода(Рис.20).



```
[sktursunboev@fedora lab05]$ ld -m elf_i386 -o lab5-2 lab5-2.o
[sktursunboev@fedora lab05]$ ./lab5-2
Введите строку:
Турсунбоев
```

Рис. 3.20: Проверка работоспособности lab5-2.asm

## 4 Выводы

Я научился работать с программой Midnight Commander