

Отчёт по лабораторной работе

Лабораторная №5

Полина Витальевна Барабаш

Содержание

| | | |
|-----------------|---|------------------|
| <i>1</i> | <i>Цель работы</i> | <i>4</i> |
| <i>2</i> | <i>Выполнение работы</i> | <i>5</i> |
| <i>3</i> | <i>Выполнение самостоятельной работы</i> | <i>17</i> |
| <i>4</i> | <i>Выводы</i> | <i>23</i> |

Список иллюстраций

| | | |
|------|---|----|
| 2.1 | Открытие Midnight Commander | 5 |
| 2.2 | Переход в каталог | 6 |
| 2.3 | Нахождение в каталоге, созданном с помощью клавиши F7 | 7 |
| 2.4 | Создание файла | 8 |
| 2.5 | Открытый в nano файл | 9 |
| 2.6 | Копирование текста программы в редактор | 10 |
| 2.7 | Проверка содержания текста программы в файле | 11 |
| 2.8 | Выполнение действий для запуска программы и взаимодействие с ней | 12 |
| 2.9 | Перемещение файла из одного каталога в другой | 13 |
| 2.10 | Копирование файла с другим именем | 14 |
| 2.11 | Текст программы с использованием подпрограмм | 15 |
| 2.12 | Выполнение программы с подпрограммами | 15 |
| 2.13 | Выполнение программы с заменённой подпрограммой | 16 |
| 3.1 | Текст программы с добавленным фрагментом для вывода введенной строки | 19 |
| 3.2 | Выполнение новой программы | 20 |
| 3.3 | Текст программы с добавленной подпрограммой для вывода введенной строки | 21 |
| 3.4 | Выполнение новой программы с подпрограммой | 22 |

1 Цель работы

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

2 Выполнение работы

Задание №1. Откройте Midnight Commander

Я открыла Midnight Commander с помощью команды mc (рис. 2.1).

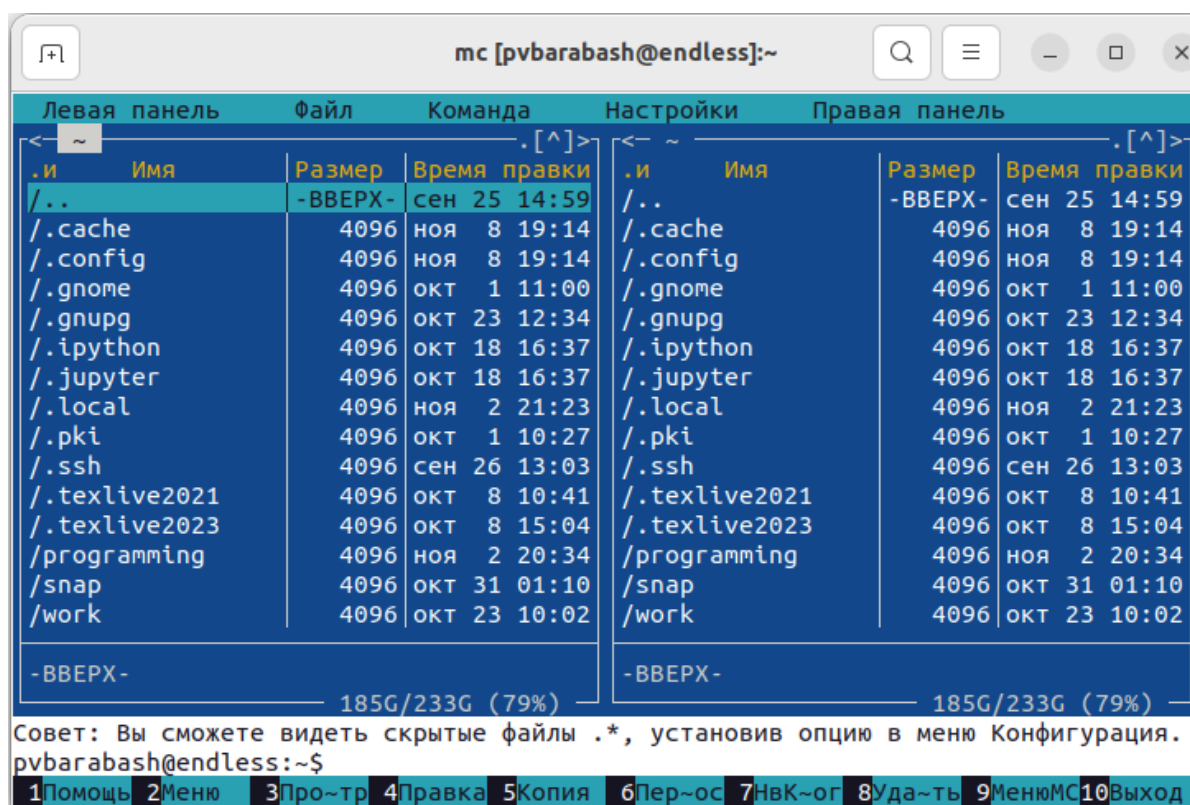


Рис. 2.1: Открытие Midnight Commander

Задание №2. Пользуясь стрелочками и клавишей Enter перейдите в каталог ~/work/arch-рс созданный при выполнении лабораторной работы №4

Я перешла в созданный каталог ~/work/arch-рс, используя управление (рис. 2.2).

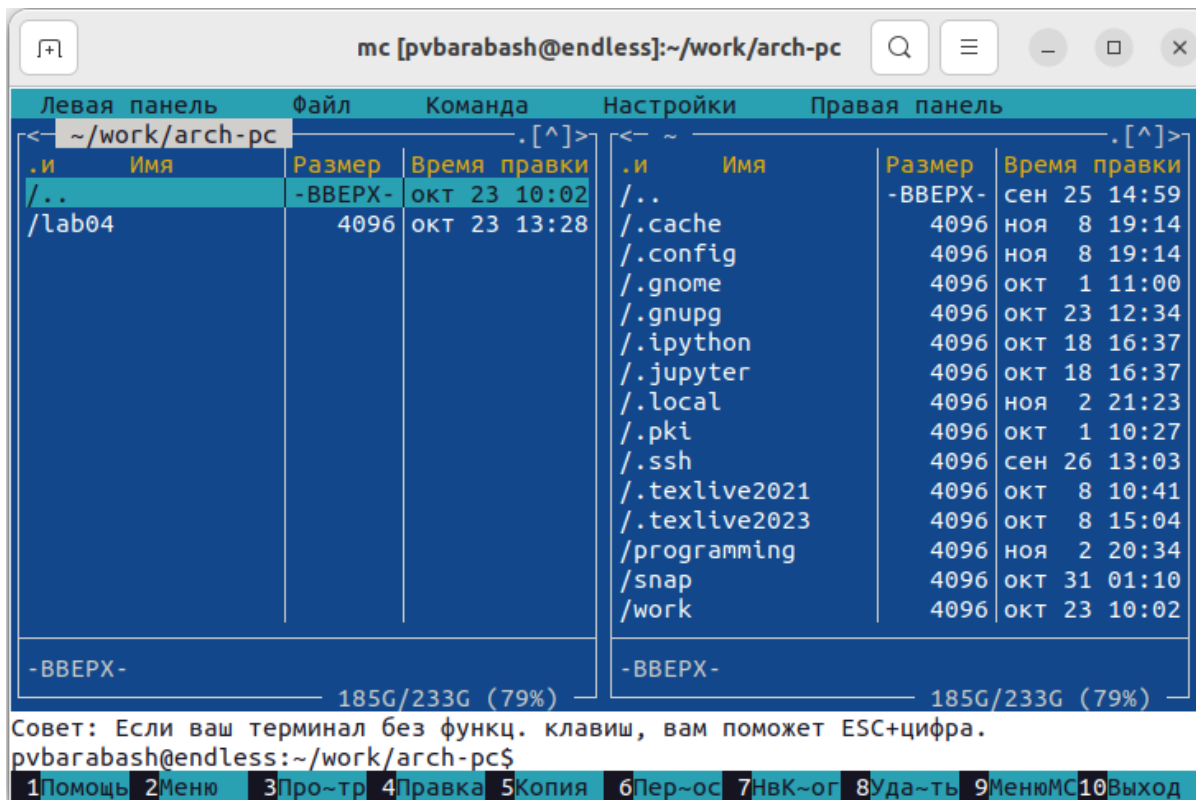
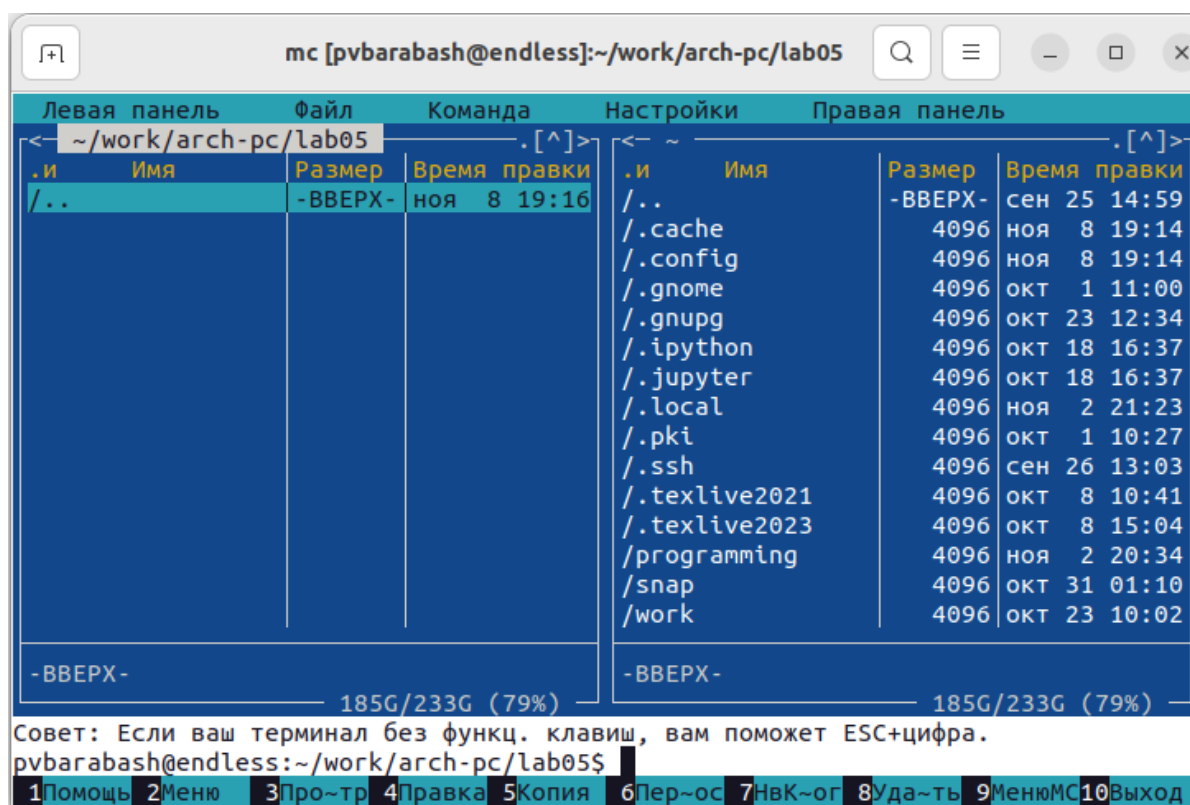


Рис. 2.2: Переход в каталог

Задание №3. С помощью функциональной клавиши F7 создайте папку lab05 и перейдите в созданный каталог.

Я создала папку с именем lab05 с помощью функциональной клавиши F7 и перешла в созданный каталог(рис. 2.3).



-ВВЕРХ-

185G/233G (79%)

-ВВЕРХ-

185G/233G (79%)

Рис. 2.3: Нахождение в каталоге, созданном с помощью клавиши F7

Задание №4. Пользуясь строкой ввода и командой touch создайте файл lab5-1.asm.

Я создала файл lab5-1.asm с помощью команды touch, которую ввела в строку ввода терминала(рис. 2.4).

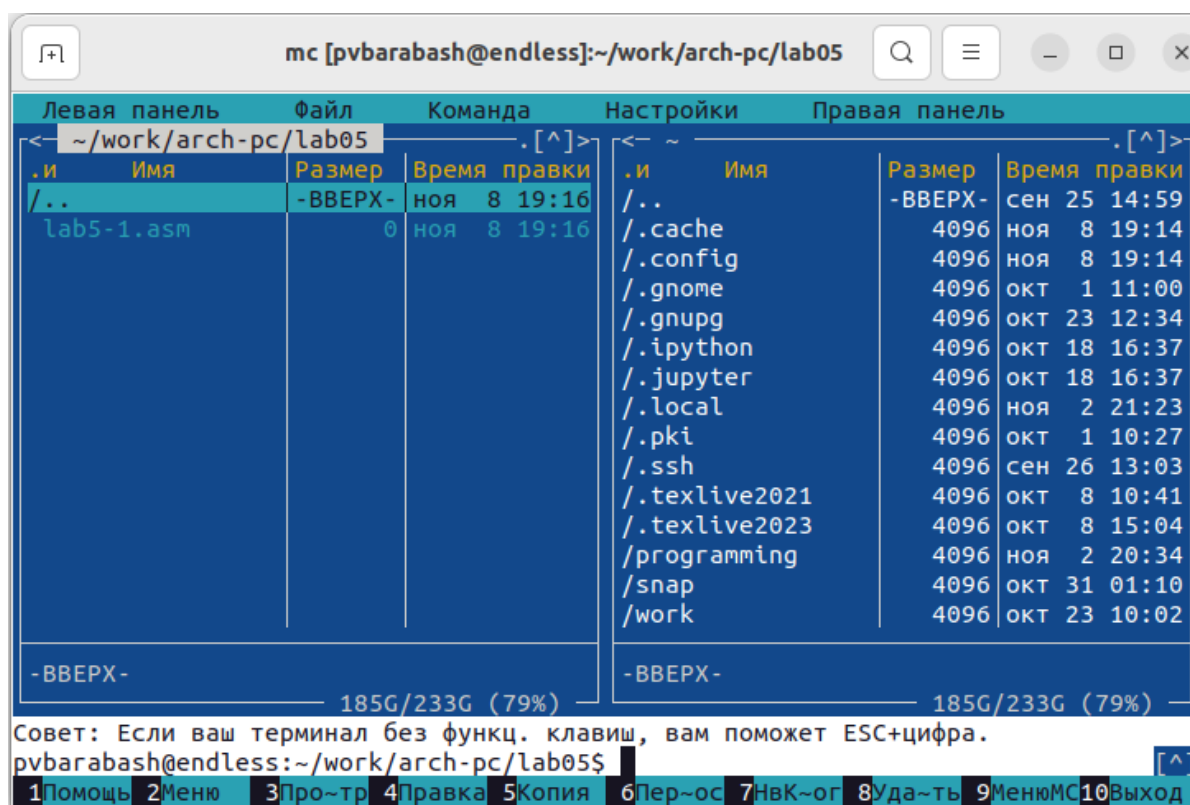


Рис. 2.4: Создание файла

Задание №5. С помощью функциональной клавиши F4 откройте файл lab5-1.asm для редактирования во встроенном редакторе.

С помощью функциональной клавиши F4 я открыла файл lab5-1.asm в редакторе nano, который выбрала из списка предложенных встроенных редакторов (рис. 2.5).

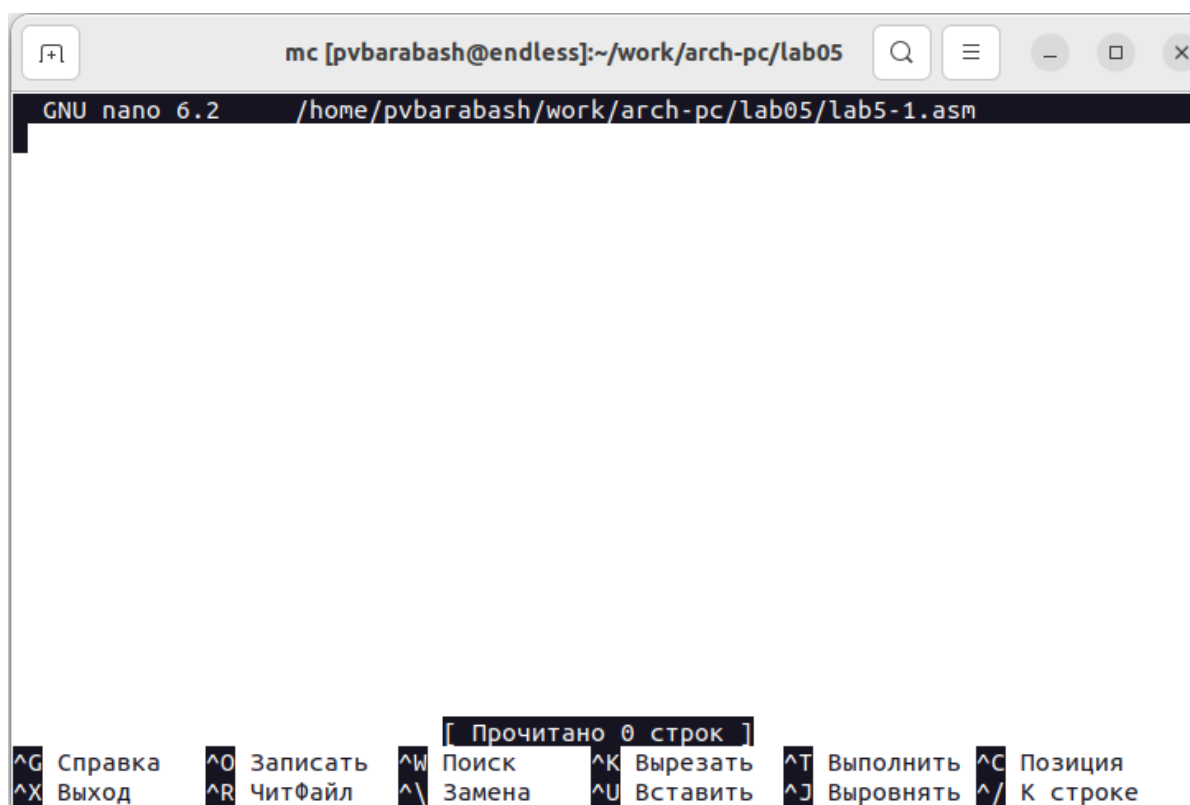


Рис. 2.5: Открытый в nano файл

Задание №6. Введите текст программы из листинга 5.1 (можно без комментариев), сохраните изменения и закройте файл.

Я ввела необходимый текст программы из листинга без комментариев и сохранила изменения с помощью приведённой последовательности действий: Ctrl + x (выход) > Y (сохранить изменения) > Enter (рис. 2.6).

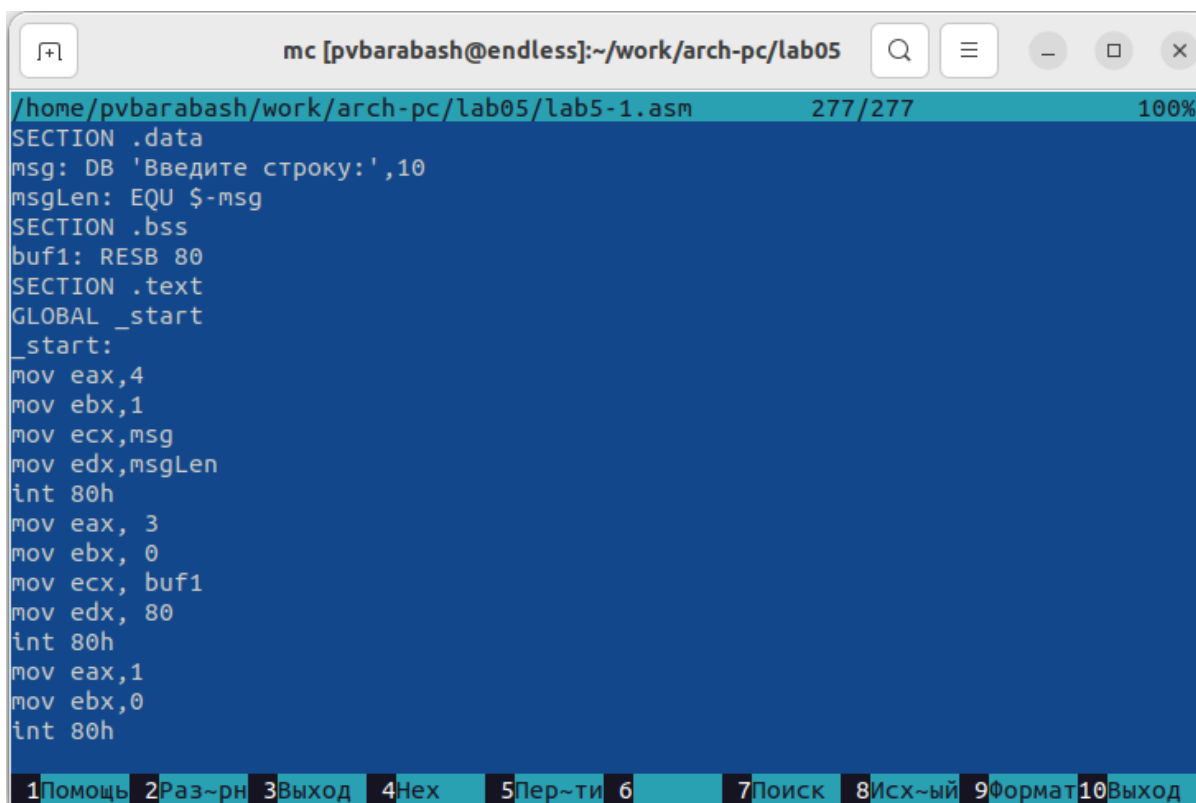
```
mc [pvbarabash@endless]:~/work/arch-pc/lab05
GNU nano 6.2 /home/pvbarabash/work/arch-pc/lab05/lab5-1.asm *
msg: DB 'Введите строку:',10
msgLen: EQU $-msg
SECTION .bss
buf1: RESB 80
SECTION .text
GLOBAL _start
_start:
mov eax,4
mov ebx,1
mov ecx,msg
mov edx,msgLen
int 80h
mov eax, 3
mov ebx, 0
mov ecx, buf1
mov edx, 80
int 80h
mov eax,1
mov ebx,0
int 80h

^G Справка      ^O Записать     ^W Поиск        ^K Вырезать     ^T Выполнить    ^С Позиция
^X Выход        ^R ЧитФайл     ^\ Замена       ^U Вставить     ^J Выровнять    ^/ К строке
```

Рис. 2.6: Копирование текста программы в редактор

Задание №7. С помощью функциональной клавиши F3 откройте файл lab5-1.asm для просмотра. Убедитесь, что файл содержит текст программы.

С помощью функциональной клавиши F3 я открыла файл lab5-1.asm для просмотра и убедилась, что файл содержит текст программы (рис. 2.7).



```
mc [pvbarabash@endless]:~/work/arch-pc/lab05
/home/pvbarabash/work/arch-pc/lab05/lab5-1.asm 277/277 100%
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,1
mov ebx,0
int 80h
1Помощь 2Раз~рн 3Выход 4Нех 5Пер~ти 6 7Поиск 8Исх~ый 9Формат10Выход
```

Рис. 2.7: Проверка содержания текста программы в файле

Задание №8. Оттранслируйте текст программы lab5-1.asm в объектный файл. Выполните компоновку объектного файла и запустите получившийся исполняемый файл. Программа выводит строку ‘Введите строку:’ и ожидает ввода с клавиатуры. На запрос введите Ваши ФИО.

Я оттранслировала файл lab5-1.asm в объектный файл с помощью `nasm -f elf lab5-1.asm`, выполнила компоновку объектного файла (`ld -m elf_i386 -o lab5-1 lab5-1.o`) и запустила получившийся исполняемый файл (`./lab5-1`). В терминале появилась строка “Введите строку:”. Я ввела своё ФИО: Барабаш Полина Витальевна (рис. 2.8).

```
pvbarabash@endless:~/work/arch-pc/lab05$ nasm -f elf lab5-1.asm
pvbarabash@endless:~/work/arch-pc/lab05$ ld -m elf_i386 -o lab5-1 lab5-1.o
pvbarabash@endless:~/work/arch-pc/lab05$ ./lab5-1
Введите строку:
Барабаш Полина Витальевна
pvbarabash@endless:~/work/arch-pc/lab05$
```

Рис. 2.8: Выполнение действий для запуска программы и взаимодействие с ней

Задание №9. В одной из панелей mc откройте каталог с файлом lab5-1.asm. В другой панели каталог со скаченным файлом in_out.asm (для перемещения между панелями используйте Tab). Скопируйте файл in_out.asm в каталог с файлом lab5-1.asm.

Открыв в одной панели каталог с файлом lab5-1.asm, а в другой каталог “Загрузки” со скаченным файлом in_out.asm, я переместила in_out.asm с помощью функциональной клавиши F6 (так как она отвечает за перемещение) (рис. 2.9).

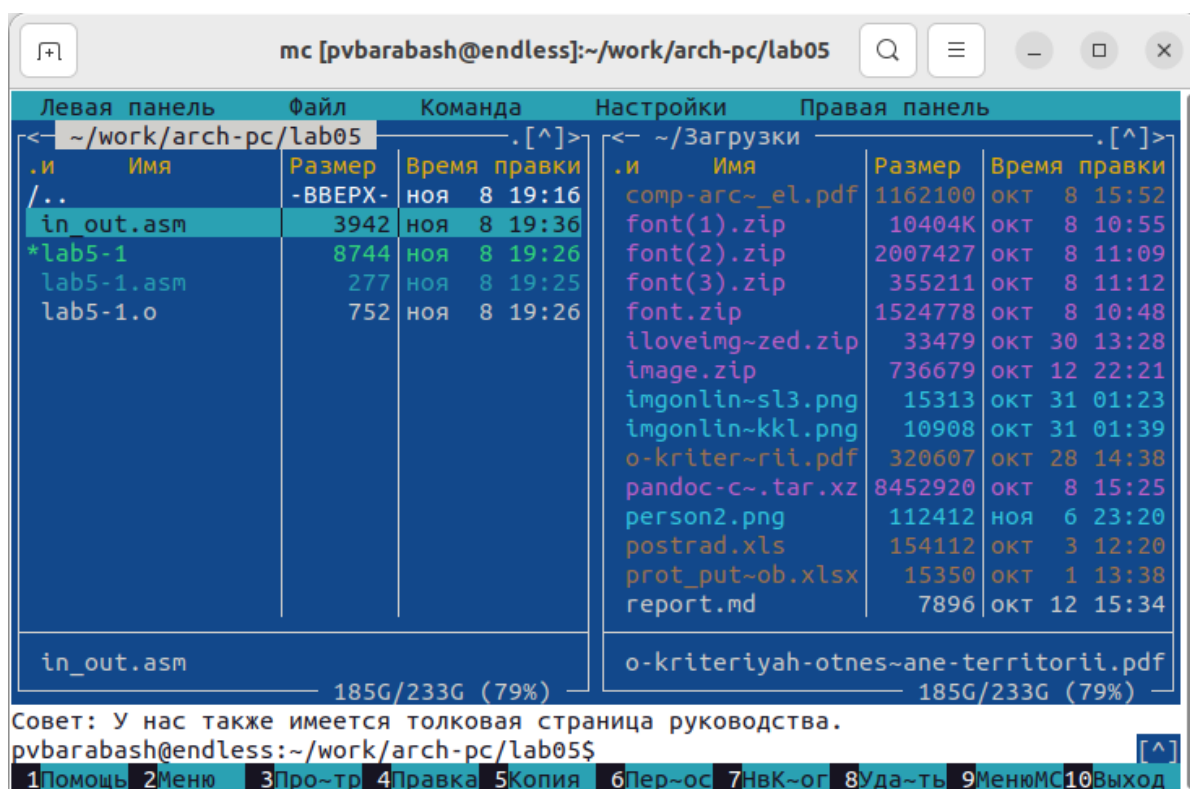


Рис. 2.9: Перемещение файла из одного каталога в другой

Задание №10. С помощью функциональной клавиши создайте копию файла lab5-1.asm с именем lab5-2.asm.

С помощью функциональной клавиши F5 (так как она отвечает за копирование) я создала копию файла lab5-1.asm с именем lab5-2.asm (рис. 2.10).

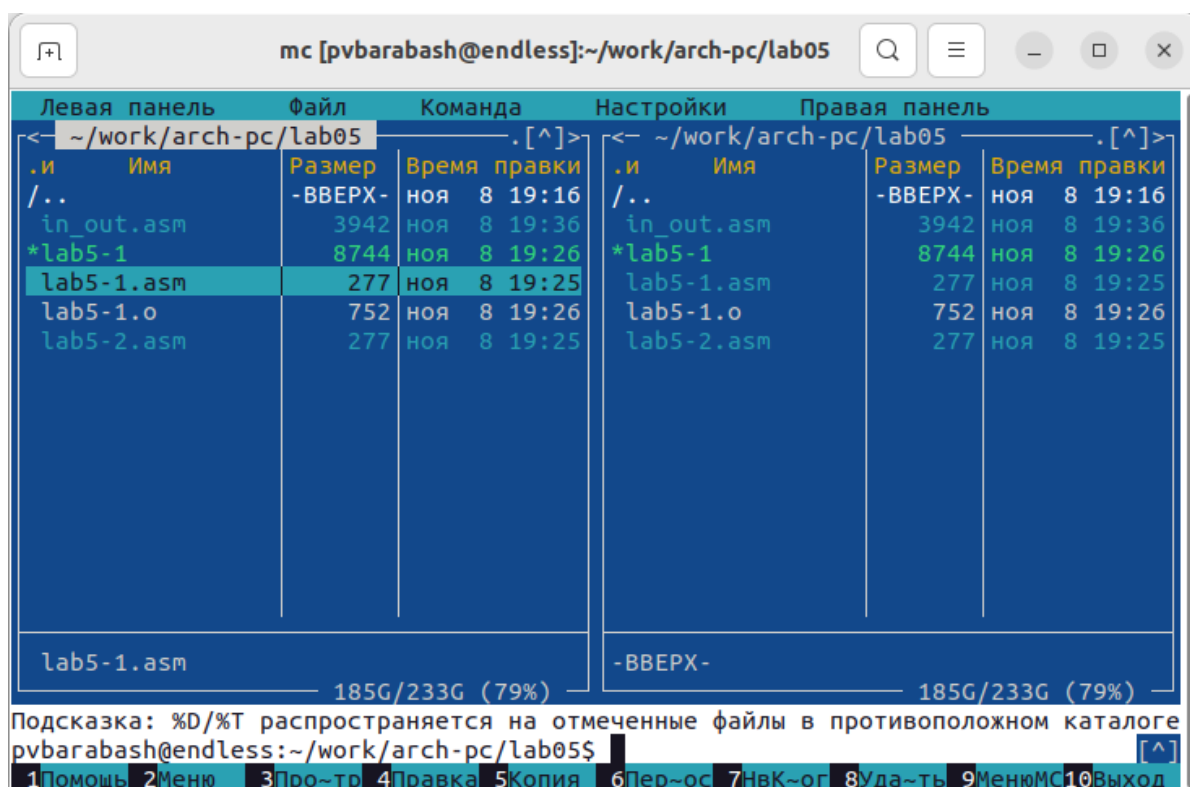


Рис. 2.10: Копирование файла с другим именем

Задание №11. Исправьте текст программы в файле lab5-2.asm с использование подпрограмм из внешнего файла in_out.asm (используйте подпрограммы sprintf, sread и quit) в соответствии с листингом 5.2. Создайте исполняемый файл и проверьте его работу.

Я исправила текст программы в lab5-2.asm с использование подпрограмм из внешнего файла in_out.asm (использовала подпрограммы sprintf, sread и quit) в соответствии с листингом 5.2. (рис. 2.11).



```
mc [pvbarabash@endless]:~/work/arch-pc/lab05
GNU nano 6.2 /home/pvbarabash/work/arch-pc/lab05/lab5-2.asm
%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

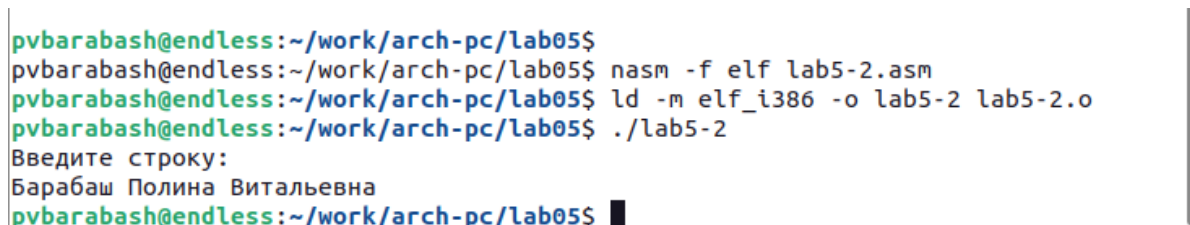
call sread
call quit
```

Прочитана 21 строка

^G Справка ^O Записать ^W Поиск ^K Вырезать ^T Выполнить ^C Позиция
^X Выход ^R ЧитФайл ^\ Замена ^U Вставить ^J Выводить ^_ К строке

Рис. 2.11: Текст программы с использованием подпрограмм

Я создала исполняемый файл и проверила его работу (рис. 2.12).



```
pvbarabash@endless:~/work/arch-pc/lab05$
pvbarabash@endless:~/work/arch-pc/lab05$ nasm -f elf lab5-2.asm
pvbarabash@endless:~/work/arch-pc/lab05$ ld -m elf_i386 -o lab5-2 lab5-2.o
pvbarabash@endless:~/work/arch-pc/lab05$ ./lab5-2
Введите строку:
Барабаш Полина Витальевна
pvbarabash@endless:~/work/arch-pc/lab05$
```

Рис. 2.12: Выполнение программы с подпрограммами

Задание № 12. В файле lab5-2.asm замените подпрограмму sprintLF на sprint. Создайте исполняемый файл и проверьте его работу. В чем разница?

Я заменила подпрограмму sprintLF на sprint. Создала исполняемый файл и проверила его работу (рис. 2.13).

```
pvbarabash@endless:~/work/arch-pc/lab05$ nasm -f elf lab5-2.asm
pvbarabash@endless:~/work/arch-pc/lab05$ ld -m elf_i386 -o lab5-2 lab5-2.o
pvbarabash@endless:~/work/arch-pc/lab05$ ./lab5-2
Введите строку: Барабаш Полина Витальевна
pvbarabash@endless:~/work/arch-pc/lab05$
```

Рис. 2.13: Выполнение программы с заменённой подпрограммой

Отличием является то, что при подпрограмме `sprintLF` после вопроса идёт перенос строки, а при подпрограмме `sprint` ответ вводится на той же строке.

3 Выполнение самостоятельной работы

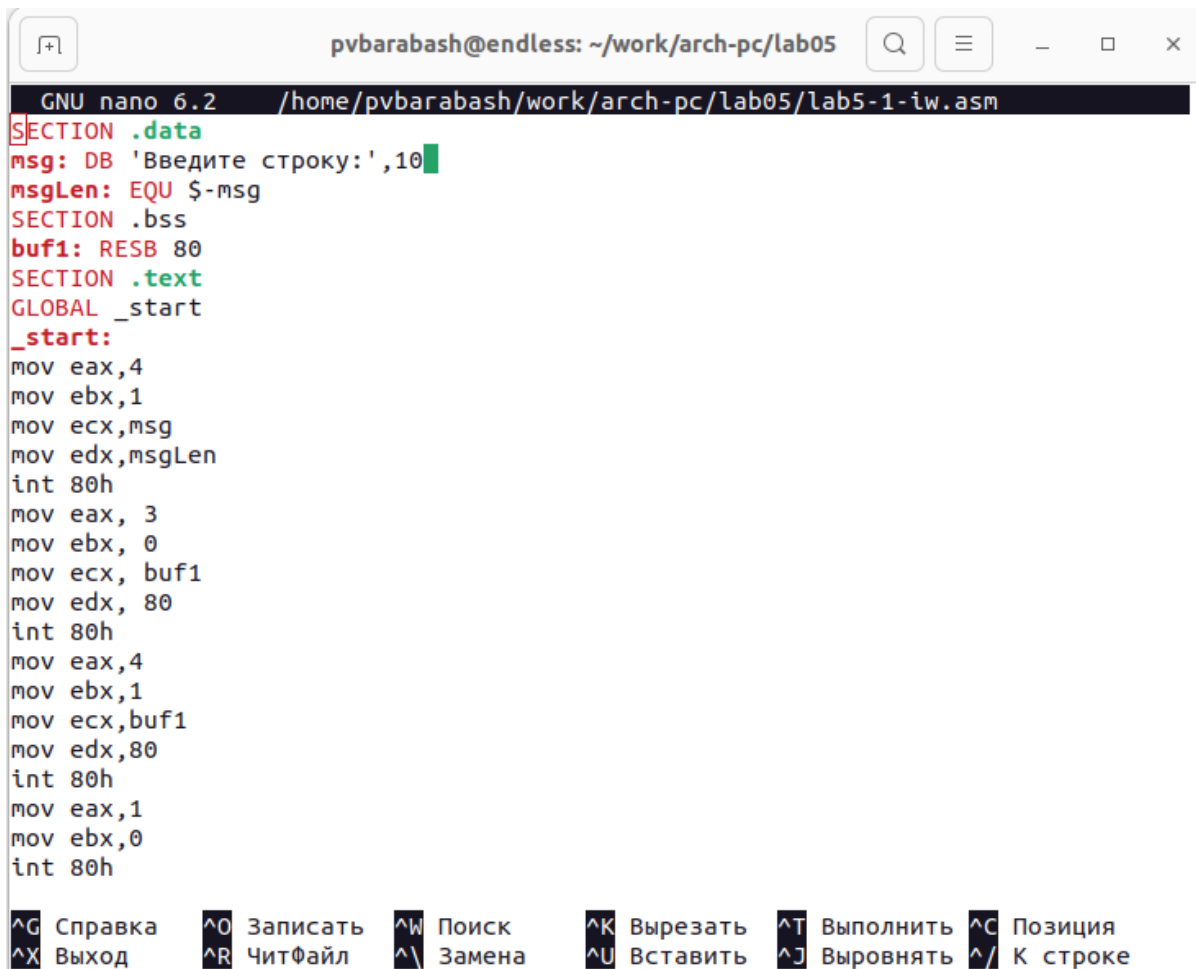
Задание №1. Создайте копию файла lab5-1.asm. Внесите изменения в программу (без использования внешнего файла in_out.asm), так чтобы она работала по следующему алгоритму:

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

С помощью функциональной клавиши F5 я создала копию файла lab5-1.asm с именем lab5-1-iw.asm и добавила в неё часть кода, чтобы программа выводила строку, введённую с клавиатуры. Текст кода, жирным выделен добавленный фрагмент:

```
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, 80
int 80h
mov eax, 1
mov ebx, 0
int 80h
(рис. 3.1).
```



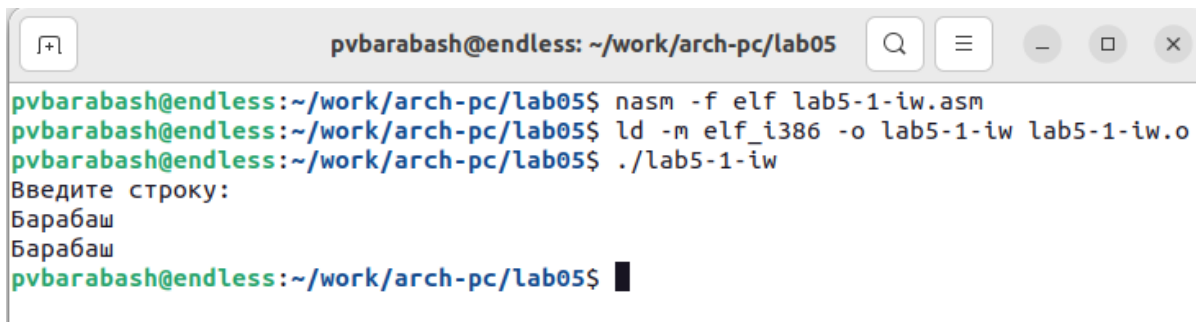
```
GNU nano 6.2 /home/pvbarabash/work/arch-pc/lab05/lab5-1-iw.asm
SECTION .data
msg: DB 'Введите строку:',10
msgLen: EQU $-msg
SECTION .bss
buf1: RESB 80
SECTION .text
GLOBAL _start
_start:
mov eax,4
mov ebx,1
mov ecx,msg
mov edx,msgLen
int 80h
mov eax, 3
mov ebx, 0
mov ecx, buf1
mov edx, 80
int 80h
mov eax,4
mov ebx,1
mov ecx,buf1
mov edx,80
int 80h
mov eax,1
mov ebx,0
int 80h

^G Справка      ^O Записать     ^W Поиск        ^K Вырезать     ^T Выполнить    ^C Позиция
^X Выход        ^R ЧитФайл     ^\ Замена       ^U Вставить     ^J Выровнять    ^/_ К строке
```

Рис. 3.1: Текст программы с добавленным фрагментом для вывода введенной строки

Задание №2. Получите исполняемый файл и проверьте его работу. На приглашение ввести строку введите свою фамилию.

Я получила исполняемый файл и проверила его работу. На приглашение ввести строку я ввела свою фамилию и программа вывела её обратно (рис. 3.2).

A screenshot of a terminal window with a title bar that reads 'pvbarabash@endless: ~/work/arch-pc/lab05'. The terminal shows the following commands and output:

```
pvbarabash@endless:~/work/arch-pc/lab05$ nasm -f elf lab5-1-iw.asm
pvbarabash@endless:~/work/arch-pc/lab05$ ld -m elf_i386 -o lab5-1-iw lab5-1-iw.o
pvbarabash@endless:~/work/arch-pc/lab05$ ./lab5-1-iw
Введите строку:
Барабаш
Барабаш
pvbarabash@endless:~/work/arch-pc/lab05$
```

Рис. 3.2: Выполнение новой программы

Задание №3. Создайте копию файла lab5-2.asm. Исправьте 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.

С помощью функциональной клавиши F5 я создала копию файла lab5-2.asm с именем lab5-2-iw.asm и исправила текст программы, добавив подпрограмму sprintLF для вывода введенной строки (на новой строке). Текст кода, жирным выделен добавленный фрагмент кода:

```
%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
mov eax, buf1
call sprintLF
call quit
(рис. 3.3).

```

```

GNU nano 6.2 /home/pvbarabash/work/arch-pc/lab05/lab-2-iw.asm
#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

mov eax, buf1
call sprintLF

call quit

```

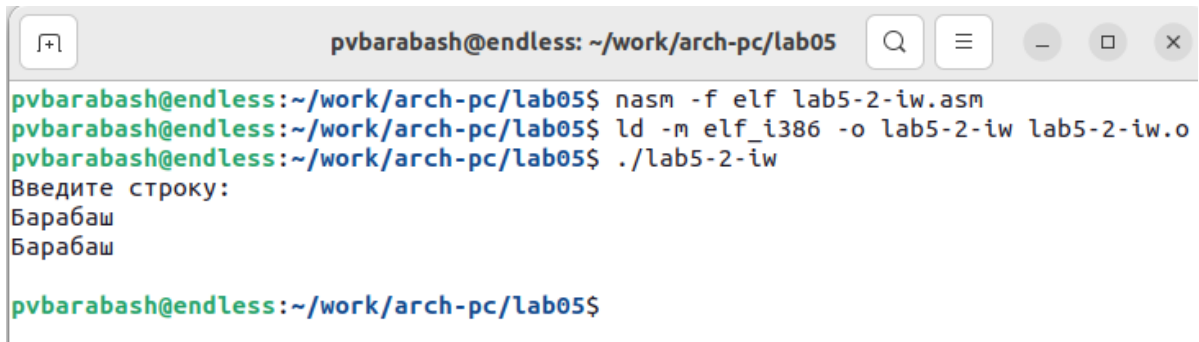
Прочитано 24 строки

^G Справка ^O Записать ^W Поиск ^K Вырезать ^T Выполнить ^C Позиция
 ^X Выход ^R ЧитФайл ^\ Замена ^U Вставить ^J Выводить ^/ К строке

Рис. 3.3: Текст программы с добавленной подпрограммой для вывода введенной строки

Задание №4. Создайте исполняемый файл и проверьте его работу.

Я создала исполняемый файл и проверила его работу. На приглашение ввести строку я ввела свою фамилию и программа вывела её обратно (рис. 3.4).



```
pvbarabash@endless: ~/work/arch-pc/lab05
pvbarabash@endless:~/work/arch-pc/lab05$ nasm -f elf lab5-2-iw.asm
pvbarabash@endless:~/work/arch-pc/lab05$ ld -m elf_i386 -o lab5-2-iw lab5-2-iw.o
pvbarabash@endless:~/work/arch-pc/lab05$ ./lab5-2-iw
Введите строку:
Барабаш
Барабаш
pvbarabash@endless:~/work/arch-pc/lab05$
```

Рис. 3.4: Выполнение новой программы с подпрограммой

4 Выводы

Я приобрела практические навыки работы в Midnight Commander. Научилась открывать Midnight Commander, передвигаться внутри него, используя стрелочки, клавиши Enter и Tab. Взаимодействовать с помощью функциональных клавиш: открывать для редактирования и для просмотра файлы, создавать каталоги, копировать и перемещать файлы. Я освоила инструкции языка ассемблера mov и int. Узнала как выводить строки, как считывать введенные строки, как завершать программы. Узнала, как подключать подпрограммы из внешнего файла.