

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных
наук Кафедра прикладной информатики и теории
вероятностей

ОТЧЕТ

ПО ЛАБОРАТОРНОЙ РАБОТЕ № 5

дисциплина: Архитектура Вычислительных Систем

Студент: Сидоренко Максим Алексеевич

Группа: НБИбд-02-22

МОСКВА

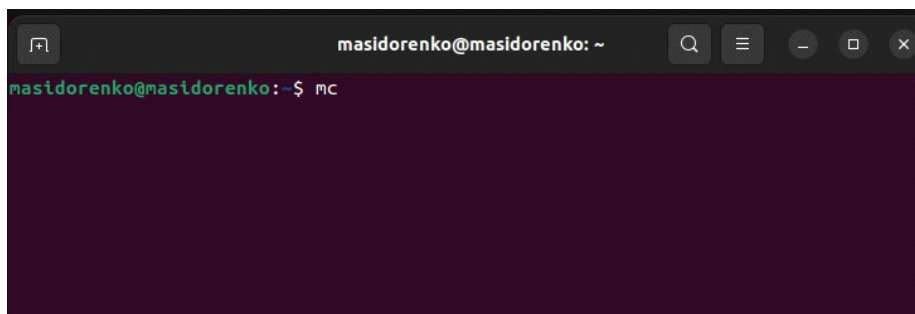
2022 г.

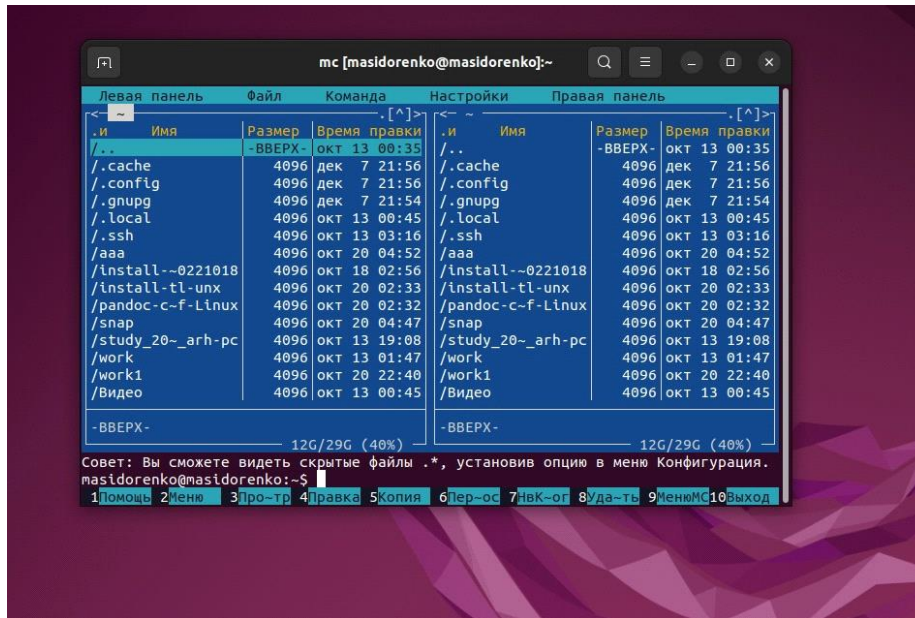
Цель работы: Приобретение практических навыков работы в Midnight Commander.

Освоение инструкций языка ассемблера `mov` и `int`.

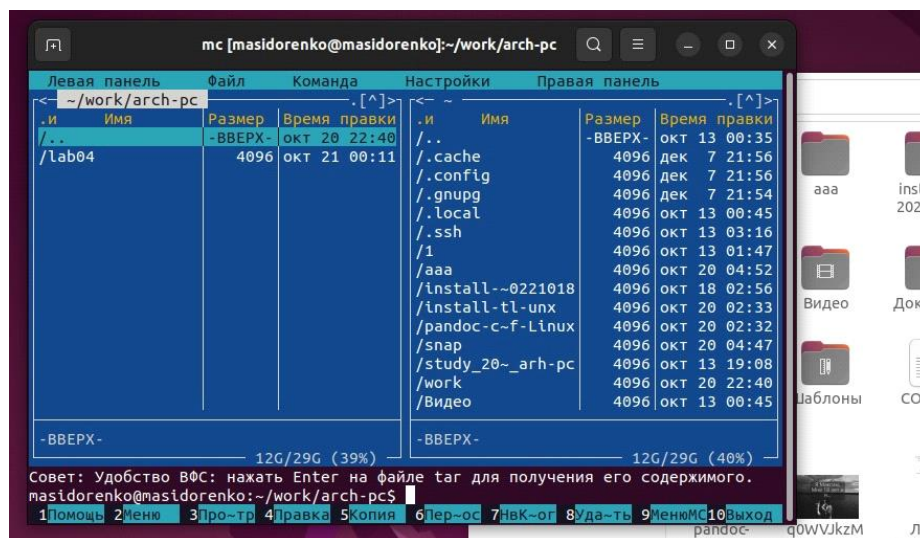
Ход работы:

- **5.3.**
- Откроем midnight commander

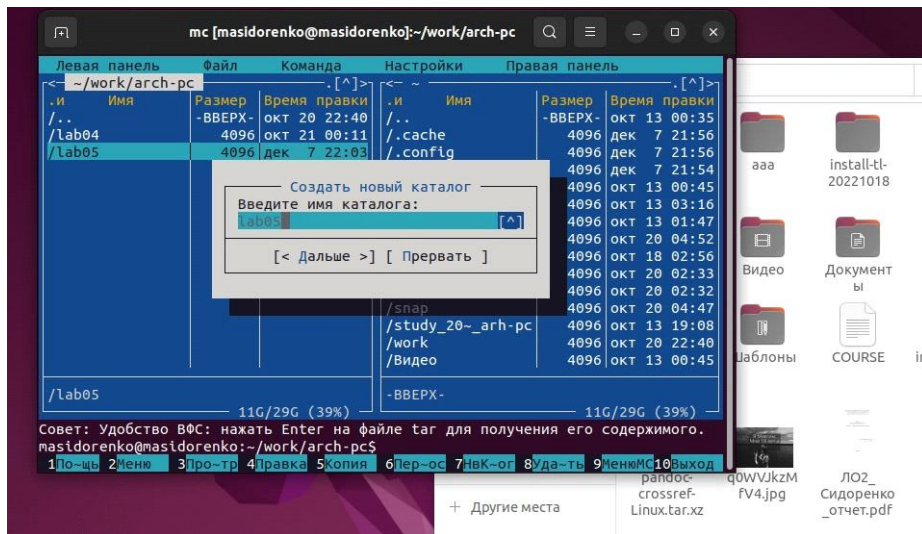




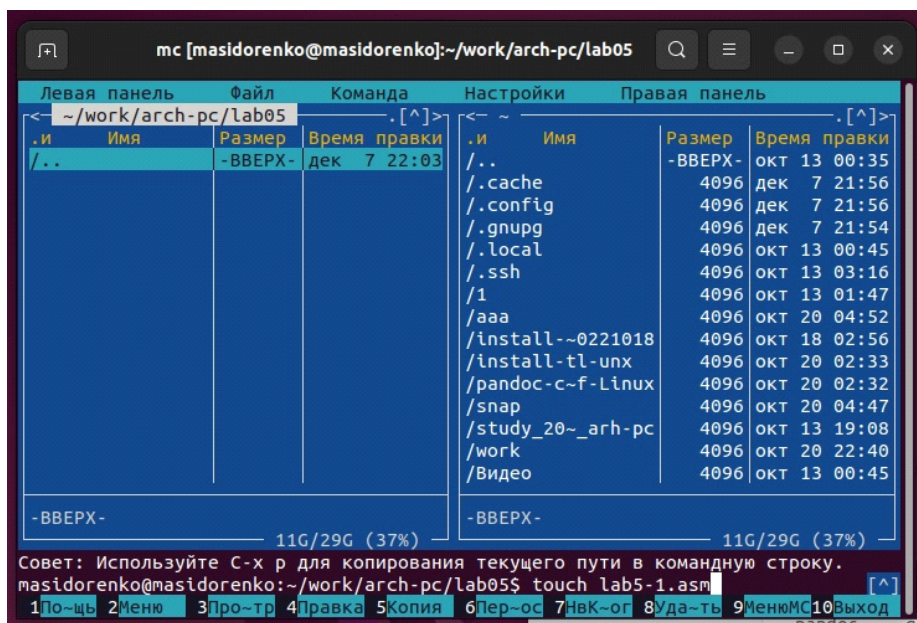
- Пользуясь клавишами “вверх вниз и ENTER” перейдем в каталог ~/work/arch-pc



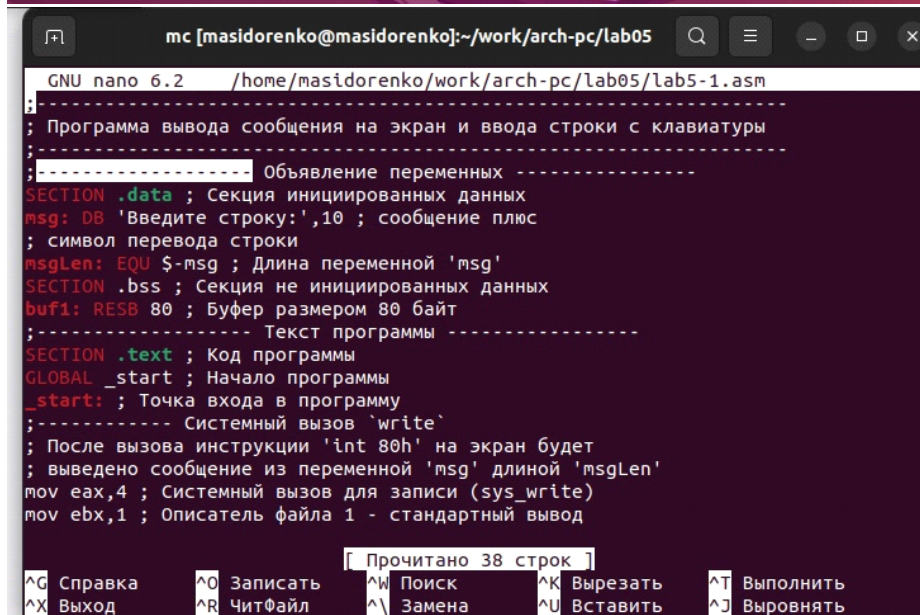
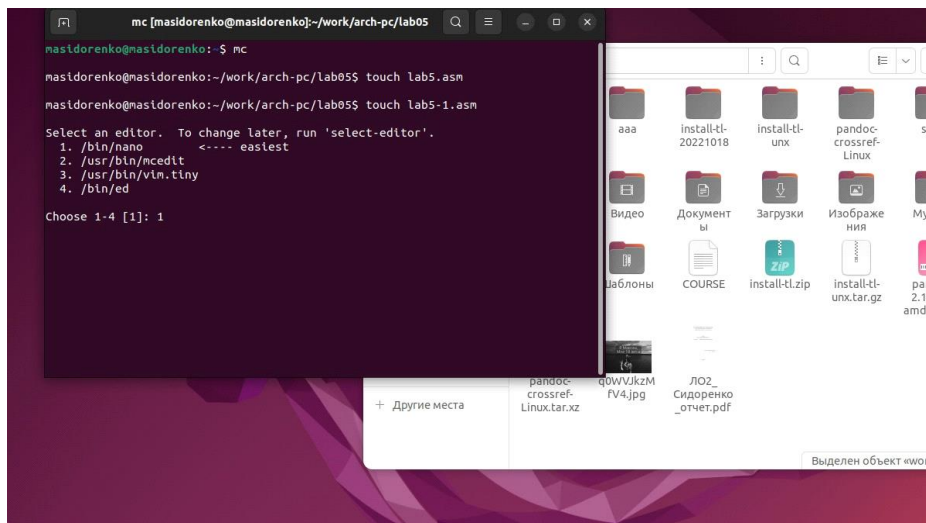
- С помощью функциональной клавиши f7 создадим папку lab05 и перейдем в созданный каталог



- Пользуясь строкой ввода и командой touch создадим файл lab5-1.asm



- С помощью функциональной клавиши F4 откроем файл lab5-1.asm для редактирования во встроенном редакторе. Как правило в качестве встроенного редактора Midnight Commander используется редактор nano



- Введем текст программы из листинга 6.1, сохраним изменения и закроем файл


```

;-----
; Программа вывода сообщения на экран и ввода строки с клавиатуры
;-----

;----- Объявление переменных -----
SECTION .data ; Секция инициализированных данных
msg: DB 'Введите строку:',10 ; сообщение плюс
; символ перевода строки

```

76

Денидова А. В.

Архитектура 386

```

msgLen: EQU $-msg ; Длина переменной 'msg'

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

;----- Текст программы -----

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

;----- Системный вызов 'write'
; После вызова инструкции 'int 80h' на экран будет
; выведено сообщение из переменной 'msg' длиной 'msgLen'

mov eax,4 ; Системный вызов для записи (sys_write)
mov ebx,1 ; Описатель файла 1 - стандартный вывод
mov ecx,msg ; Адрес строки 'msg' в 'ecx'
mov edx,msgLen ; Размер строки 'msg' в 'edx'
int 80h ; Вызов ядра

;----- системный вызов 'read' -----
; После вызова инструкции 'int 80h' программа будет ожидать ввода
; строки, которая будет записана в переменную 'buf1' размером 80
; байт

mov eax,3 ; Системный вызов для чтения (sys_read)
mov ebx,0 ; Дескриптор файла 0 - стандартный ввод
mov ecx,buf1 ; Адрес буфера под вводимую строку
mov edx,80 ; Длина вводимой строки
int 80h ; Вызов ядра

```

```

mc [masidorenko@masidorenko]:~/work/arch-pc/lab05
GNU nano 6.2 /home/masidorenko/work/arch-pc/lab05/lab5-1.asm
;-----
; Программа вывода сообщения на экран и ввода строки с клавиатуры
;-----
;----- Объявление переменных -----
SECTION .data ; Секция инициализированных данных
msg: DB 'Введите строку:',10 ; сообщение плюс
; символ перевода строки
msgLen: EQU $-msg ; Длина переменной 'msg'
SECTION .bss ; Секция не инициализированных данных
buf1: RESB 80 ; Буфер размером 80 байт
;----- Текст программы -----
SECTION .text ; Код программы
GLOBAL _start ; Начало программы
_start: ; Точка входа в программу
;----- Системный вызов 'write'
; После вызова инструкции 'int 80h' на экран будет
; выведено сообщение из переменной 'msg' длиной 'msgLen'
mov eax,4 ; Системный вызов для записи (sys_write)
mov ebx,1 ; Описатель файла 1 - стандартный вывод
mov ecx,msg ; Адрес строки 'msg' в 'ecx'
mov edx,msgLen ; Размер строки 'msg' в 'edx'
int 80h ; Вызов ядра
;----- системный вызов 'read' -----
; После вызова инструкции 'int 80h' программа будет ожидать ввода
; строки, которая будет записана в переменную 'buf1' размером 80
; байт
mov eax,3 ; Системный вызов для чтения (sys_read)
mov ebx,0 ; Дескриптор файла 0 - стандартный ввод
mov ecx,buf1 ; Адрес буфера под вводимую строку
mov edx,80 ; Длина вводимой строки
int 80h ; Вызов ядра

Прочитано 38 строк
^G Справка      ^O Записать    ^W Поиск       ^K Вырезать    ^T Выполнить
^X Выход        ^R ЧитФайл    ^\ Замена      ^U Вставить    ^J Выровнять

```

- С помощью функциональной клавиши F3 откроем файл lab5-1.asm для просмотра, убедимся, что файл содержит текст

```

/home/masidorenko/work-ch-pc/lab05/lab5-1.asm 1419/2437 58%
;-----
; Программа вывода сообщения на экран и ввода строки с клавиатуры
;-----
;----- Объявление переменных -----
SECTION .data ; Секция инициализированных данных
msg: DB 'Введите строку:',10 ; сообщение плюс
; символ перевода строки
msgLen: EQU $-msg ; Длина переменной 'msg'
SECTION .bss ; Секция не инициализированных данных
buf1: RESB 80 ; Буфер размером 80 байт
;----- Текст программы -----
SECTION .text ; Код программы
GLOBAL _start ; Начало программы
_start: ; Точка входа в программу
;----- Системный вызов 'write'
; После вызова инструкции 'int 80h' на экран будет
; выведено сообщение из переменной 'msg' длиной 'msgLen'
mov eax,4 ; Системный вызов для записи (sys_write)
mov ebx,1 ; Описатель файла 1 - стандартный вывод

mov ecx,msg ; Адрес строки 'msg' в 'ecx'
mov edx,msgLen ; Размер строки 'msg' в 'edx'

```

программы

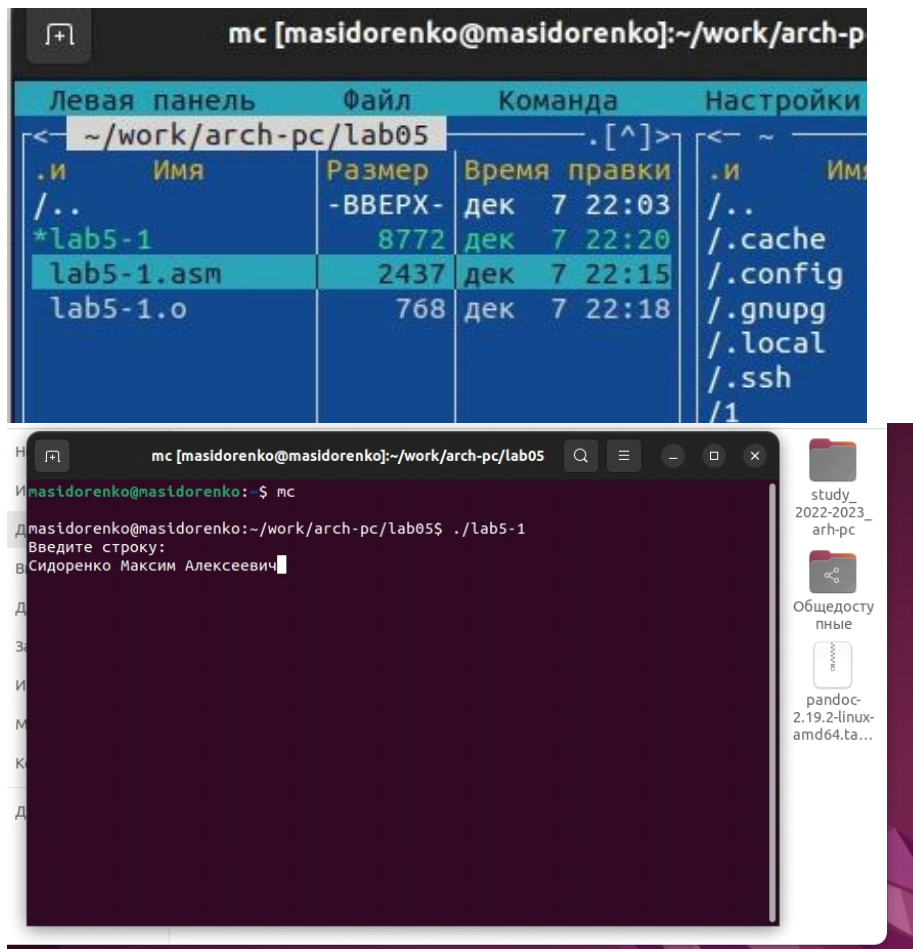
- Оттранслируем текст программы lab5-1.asm в объектный файл. Выполнили компоновку объектного файла и запустили получившийся исполняемый файл. Программа выводит строку 'Введите строку' и ожидает ввода с клавиатуру, На запрос введем ФИО

```

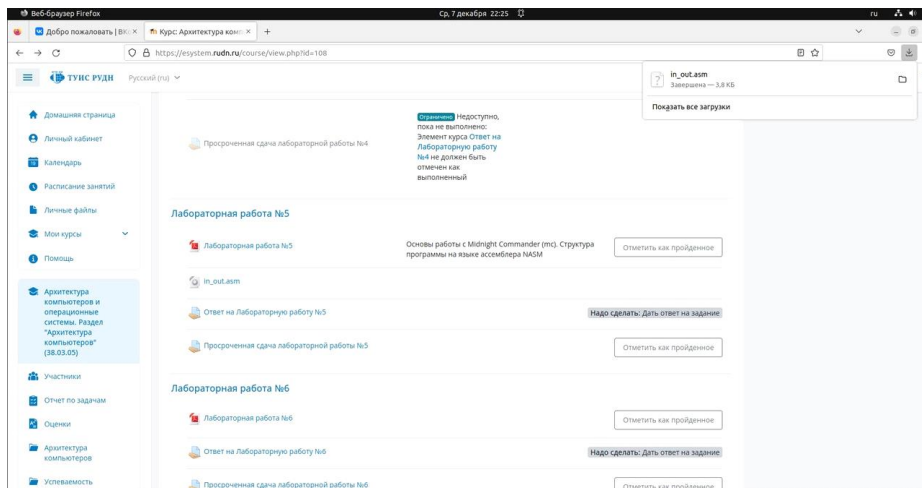
Совет: Используйте C-x p для копирования текущего пути в командную строку.
masidorenko@masidorenko:~/work/arch-pc/lab05$ nasm -f elf lab5-1.asm
1По-щъ 2Меню 3Про-тр 4Правка 5Копия 6Пер-ос 7НвК-ог 8Уда-ть 9МенюМС 10Выход

Совет: Используйте C-x p для копирования текущего пути в командную строку.
masidorenko@masidorenko:~/work/arch-pc/lab05$ ./lab5-1.o
1По-щъ 2Меню 3Про-тр 4Правка 5Копия 6Пер-ос 7НвК-ог 8Уда-ть 9МенюМС 10Выход

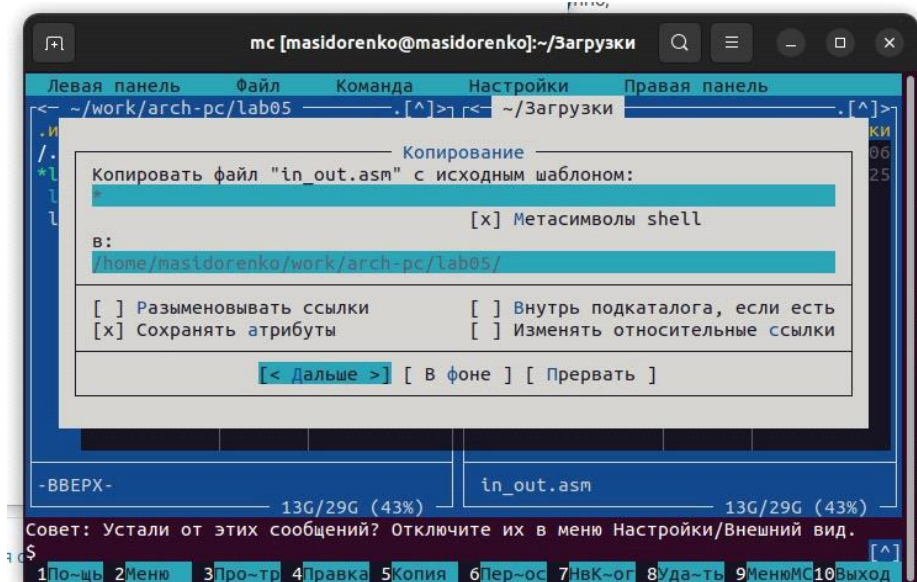
```



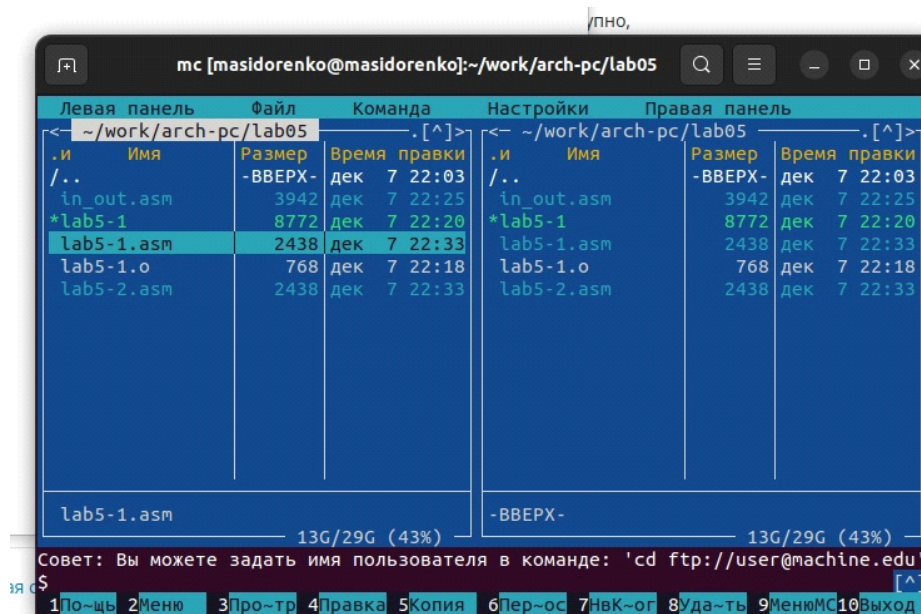
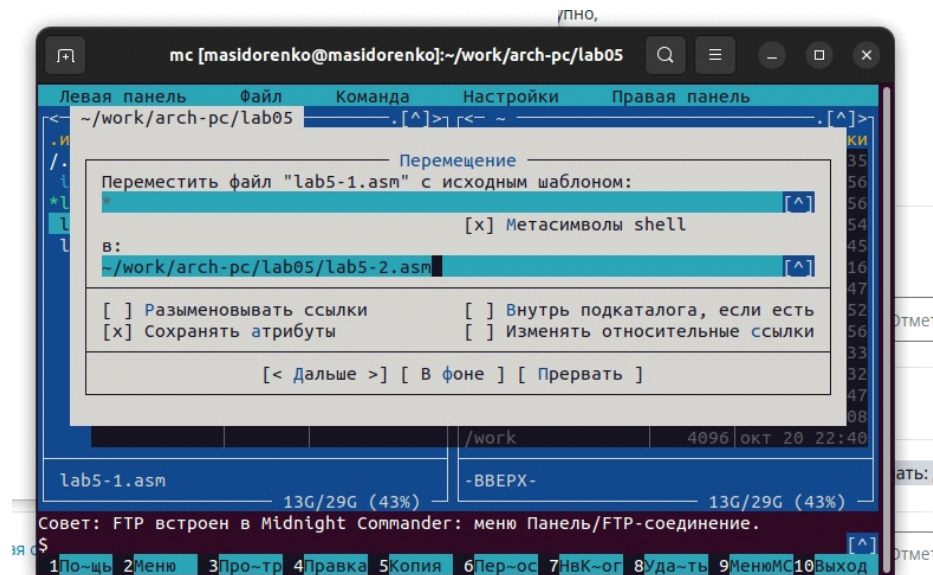
- 5.3.1. Подключение внешнего файла in_out.asm
- Скачаем файл in_out.asm со страницы курса ТУИС



- Подключаемый файл `in_out.asm` должен лежать в том же каталоге, что и файл с программой, в которой он используется. В одной из панелей `mc` откроем каталог с файлом `lab5-1.asm`. В другой панели каталог с `in_out.asm`. Скопируем файл `in_out.asm` в каталог с файлом `lab5-1.asm`



- С помощью функциональной клавиши `F6` создадим копию файла `lab5-1.asm` с именем `lab5-2.asm`. Выделим файл `lab5-1.asm`, нажмем на клавишу `F6`, введем имя файла `lab5-2.asm` и нажмем на `ENTER`



Задание №6

- Исправим текст программы в файле lab5-2.asm с использованием подпрограммы из внешнего файла in_out.asm(используем подпрограммы sprintLF, sread и quit) в соответствии с листингом 6.2. Создадим исполняемый файл и

Листинг 6.2. Программа вывода сообщения на экран и ввода строки с клавиатуры с использованием файла `in_out.asm`

```

;-----
; Программа вывода сообщения на экран и ввода строки с клавиатуры
;-----

```

```

#include 'in_out.asm' ; подключение внешнего файла

```

Демидова А. В.

81

Архитектура ЭВМ

```

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

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

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

mov eax, msg ; запись адреса выводимого сообщения в `EAX`
call sprintf ; вызов подпрограммы печати сообщения

mov ecx, buf1 ; запись адреса переменной в `EAX`
mov edx, 80 ; запись длины вводимого сообщения в `EBX`
call sread ; вызов подпрограммы ввода сообщения

call quit ; вызов подпрограммы завершения

```

проверим его работу

```

mc [masidorenko@masidorenko]:~/work/arch-pc/lab05
GNU nano 6.2 /home/masidorenko/work/arch-pc/lab05/lab5-2.asm *
;-----
; Программа вывода сообщения на экран и ввода строки с клавиатуры
;-----
#include 'in_out.asm' ; подключение внешнего файла
SECTION .data ; Секция иницированных данных
msg: DB 'Введите строку: ',0h ; сообщение
SECTION .bss ; Секция не иницированных данных
buf1: RESB 80 ; Буфер размером 80 байт
SECTION .text ; Код программы
GLOBAL _start ; Начало программы
_start: ; Точка входа в программу
mov eax, msg ; запись адреса выводимого сообщения в `EAX`
call sprintf ; вызов подпрограммы печати сообщения
mov ecx, buf1 ; запись адреса переменной в `EAX`
mov edx, 80 ; запись длины вводимого сообщения в `EBX`
call sread ; вызов подпрограммы ввода сообщения
call quit ; вызов подпрограммы завершения

```

[^]G Справка [^]O Записать [^]W Поиск [^]K Вырезать [^]T Выполнить
[^]X Выход [^]R ЧитФайл [^]\ Замена [^]U Вставить [^]J Выровнять


```
13G/29G (43%)
Совет: Вы можете выбрать редактор для F4 с помощью переменной оболочки EDITOR.
$ nasm -f elf lab5-2.asm
1По-ць 2Меню 3Про-тр 4Правка 5Копия 6Пер-ос 7НвК-ог 8Уда-ть 9МенюМС10Выход

-ВВЕРХ-
13G/29G (43%)
Совет: Вы можете выбрать редактор для F4 с помощью переменной оболочки EDITOR.
$ ld -m elf_i386 -o lab5-2 lab5-2.o
1По-ць 2Меню 3Про-тр 4Правка 5Копия 6Пер-ос 7НвК-ог 8Уда-ть 9МенюМС10Выход

1 из 343
Поделиться · Удалить · Ещё ▾

mc [masidorenko@masidorenko]:~/work/arch-pc/lab05
Левая панель
Файл
Команда
Настройки
Правая панель
<- ~/[^]>
.и Имя Размер Время правки
/.. -ВВЕРХ- дек 7 22:03
in_out.asm 3942 дек 7 22:25
*lab5-1 8772 дек 7 22:20
lab5-1.asm 2438 дек 7 22:33
lab5-1.o 768 дек 7 22:18
*lab5-2 9092 дек 7 22:59
lab5-2.asm 1226 дек 7 22:57
lab5-2.o 1312 дек 7 22:57
-ВВЕРХ-
13G/29G (43%)
Совет: Вы можете выбрать редактор для F4 с помощью переменной оболочки EDITOR.
$
1По-ць 2Меню 3Про-тр 4Правка 5Копия 6Пер-ос 7НвК-ог 8Уда-ть 9МенюМС10Выход

mc [masidorenko@masidorenko]:~/work/arch-pc/lab05
Команда «call» не найдена. Возможно, вы имели в виду:
  command 'pall' from snap pall (3.0.1)
  command 'calc' from deb calc (2.12.7.2-4)
  command 'wall' from deb bsduutils (1:2.37.2-4ubuntu3)
  command 'cal' from deb ncal (12.1.7+nmu3ubuntu2)
See 'snap info <snapname>' for additional versions.
masidorenko@masidorenko:~/work/arch-pc/lab05$
masidorenko@masidorenko:~/work/arch-pc/lab05$ cd
masidorenko@masidorenko:~$ mc
$ ./lab5-1
Введите строку:
^[^
$ ~/work/arch-pc/lab05
/bin/bash: строка 1: /home/masidorenko/work/arch-pc/lab05: Это каталог
$ nasm -f elf lab5-2.asm
$ ld -m elf_i386 -o lab5-2 lab5-2.o
$ ./lab5-2
Введите строку:
masidorenko@masidorenko
$ ./lab5-2
Введите строку:
Максим Сидоренко
```

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

Между ними видна разница в том, что sprintLF строку для ввода переносит вниз, а sprint на той же строке, где и указание 'Введите строку'

```
mc [masidorenko@masidorenko]:~/work/arch-pc/lab05
GNU nano 6.2 /home/masidorenko/work/arch-pc/lab05/lab5-2.asm *
;-----
; Программа вывода сообщения на экран и ввода строки с клавиатуры
;-----
#include 'in_out.asm' ; подключение внешнего файла
SECTION .data ; Секция инициализированных данных
msg: DB 'Введите строку: ',0h ; сообщение
SECTION .bss ; Секция не инициализированных данных
buf1: RESB 80 ; Буфер размером 80 байт
12 SECTION .text ; Код программы
GLOBAL _start ; Начало программы
_start: ; Точка входа в программу
mov eax, msg ; запись адреса выводимого сообщения в `EAX`
call sprint ; вызов подпрограммы печати сообщения
mov ecx, buf1 ; запись адреса переменной в `EAX`
mov edx, 80 ; запись длины вводимого сообщения в `EBX`
call sread ; вызов подпрограммы ввода сообщения
call quit ; вызов подпрограммы завершения

^G Справка      ^O Записать     ^M Поиск        ^K Вырезать     ^T Выполнить
^X Выход        ^R ЧитФайл     ^_ Замена      ^U Вставить     ^J Выводить
```

Демидова А. В.

81

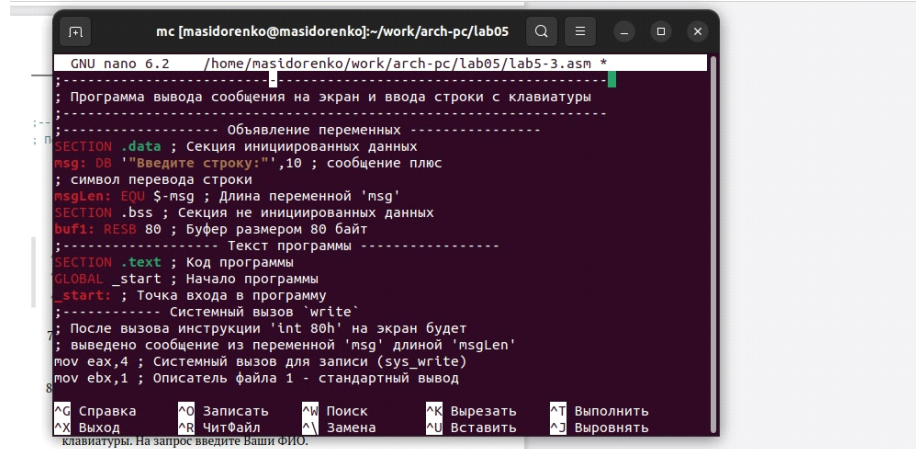
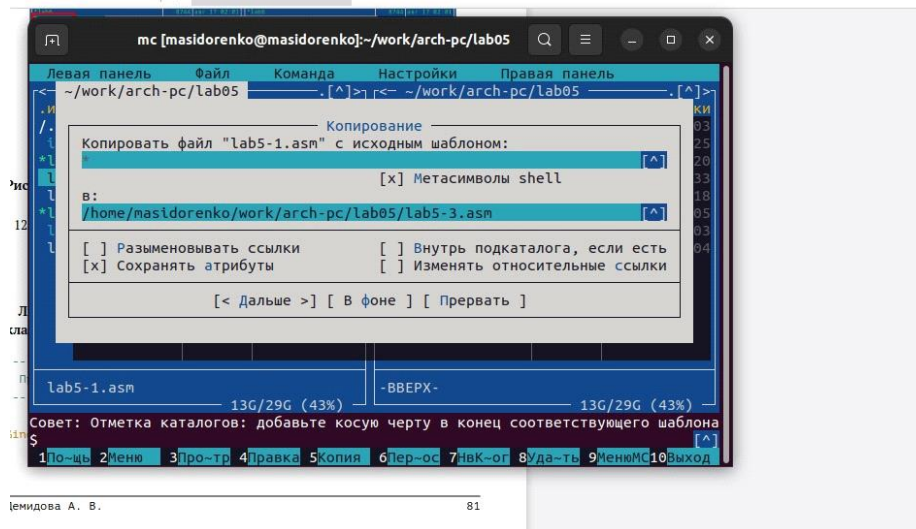
```
masidorenko@masidorenko:~/work/arch-pc/lab05$
masidorenko@masidorenko:~/work/arch-pc/lab05$ cd
masidorenko@masidorenko:~/work/arch-pc/lab05$ mc
$ ./lab5-1
Введите строку:
^[[
$ ./work/arch-pc/lab05
/bin/bash: строка 1: /home/masidorenko/work/arch-pc/lab05: Это каталог
$ nasm -f elf lab5-2.asm
12 $ ld -m elf_i386 -o lab5-2 lab5-2.o
$ ./lab5-2
Введите строку:
masidorenko@masidorenko
$ ./lab5-2
Введите строку:
Максим Сидоренко
$ ./lab5-2
Введите строку:
Максим Сидоренко
$ nasm -f elf lab5-2.asm
$ ld -m elf_i386 -o lab5-2 lab5-2.o
$ ./lab5-2
Введите строку: Максим Сидоренко
```

Демидова А. В.

81

- 5.4. Самостоятельная работа
- Создайте копию файла lab6-1.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 введенную строку на экран.



- Получим исполняемый файл и проверим его работу. На приглашение введем строку своё имя и фамилию

mc [masidorenko@masidorenko]:~/work/arch-pc/lab05

Левая панель			Файл	Команда	Настройки	Правая панель		
~/work/arch-pc/lab05			.[^]>			~/work/arch-pc/lab05		
.и	Имя	Размер	Время	правки	.и	Имя	Размер	Время
		-ВВЕРХ-	дек	7 22:03			-ВВЕРХ-	дек
	in_out.asm	3942	дек	7 22:25		in_out.asm	3942	дек
*	lab5-1	8772	дек	7 22:20	*	lab5-1	8772	дек
	lab5-1.asm	2438	дек	7 22:33		lab5-1.asm	2438	дек
	lab5-1.o	768	дек	7 22:18		lab5-1.o	768	дек
*	lab5-2	9092	дек	7 23:05	*	lab5-2	9092	дек
	lab5-2.asm	1224	дек	7 23:03		lab5-2.asm	1224	дек
	lab5-2.o	1312	дек	7 23:04		lab5-2.o	1312	дек
	lab5-3.asm	2440	дек	7 23:21		lab5-3.asm	2440	дек
lab5-3.asm			-ВВЕРХ-					
13G/29G (43%)			13G/29G (43%)					

Совет: Отметка каталогов: добавьте косую черту в конец соответствующего шаблона
\$ nasm -f elf lab5-3.asm

1По-щъ 2Меню 3Про-тр 4Правка 5Копия 6Пер-ос 7НвК-ог 8Уда-ть 9МенюМС 10Выход

клавиатуры. На запрос введите Ваши ФИО.

mc [masidorenko@masidorenko]:~/work/arch-pc/lab05

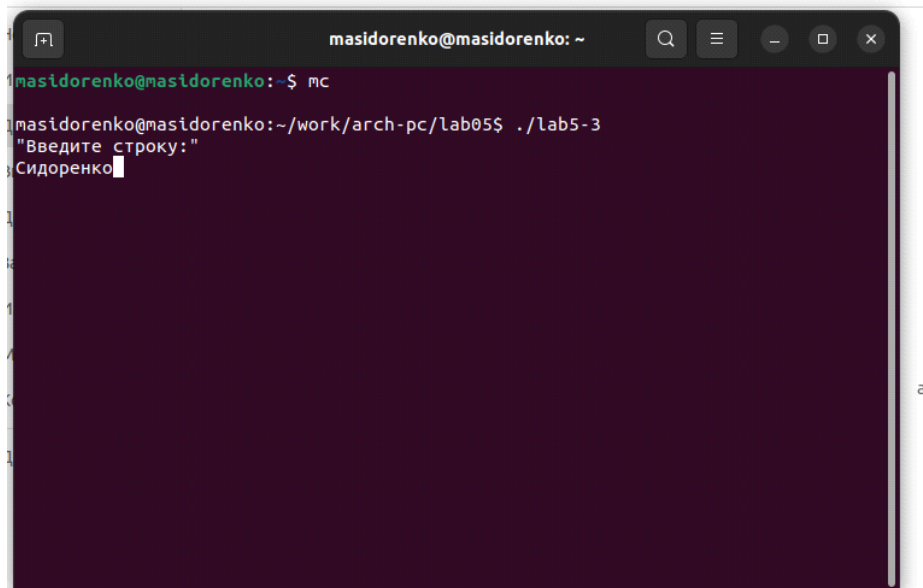
Левая панель			Файл	Команда	Настройки	Правая панель		
~/work/arch-pc/lab05			.[^]>			~/work/arch-pc/lab05		
.и	Имя	Размер	Время	правки	.и	Имя	Размер	Время
		-ВВЕРХ-	дек	7 22:03			-ВВЕРХ-	дек
	in_out.asm	3942	дек	7 22:25		in_out.asm	3942	дек
*	lab5-1	8772	дек	7 22:20	*	lab5-1	8772	дек
	lab5-1.asm	2438	дек	7 22:33		lab5-1.asm	2438	дек
	lab5-1.o	768	дек	7 22:18		lab5-1.o	768	дек
*	lab5-2	9092	дек	7 23:05	*	lab5-2	9092	дек
	lab5-2.asm	1224	дек	7 23:03		lab5-2.asm	1224	дек
	lab5-2.o	1312	дек	7 23:04		lab5-2.o	1312	дек
	lab5-3.asm	2440	дек	7 23:21		lab5-3.asm	2440	дек
	lab5-3.o	768	дек	7 23:21		lab5-3.o	768	дек
lab5-3.asm			-ВВЕРХ-					
13G/29G (43%)			13G/29G (43%)					

Совет: Отметка каталогов: добавьте косую черту в конец соответствующего шаблона
\$ ld -m elf_i386 -o lab5-3 lab5-3.o

1По-щъ 2Меню 3Про-тр 4Правка 5Копия 6Пер-ос 7НвК-ог 8Уда-ть 9МенюМС 10Выход

клавиатуры. На запрос введите Ваши ФИО.

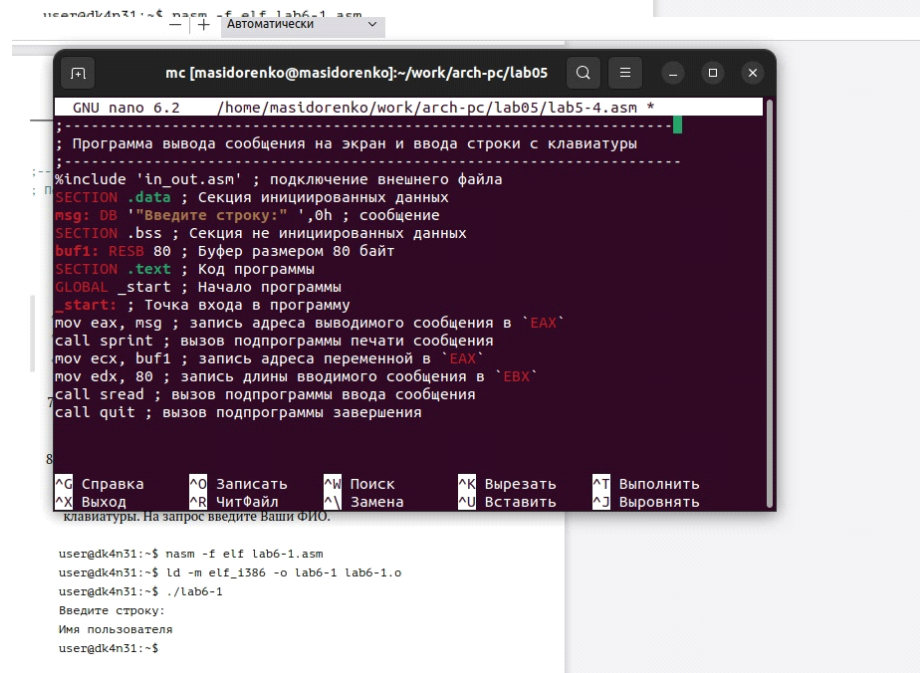
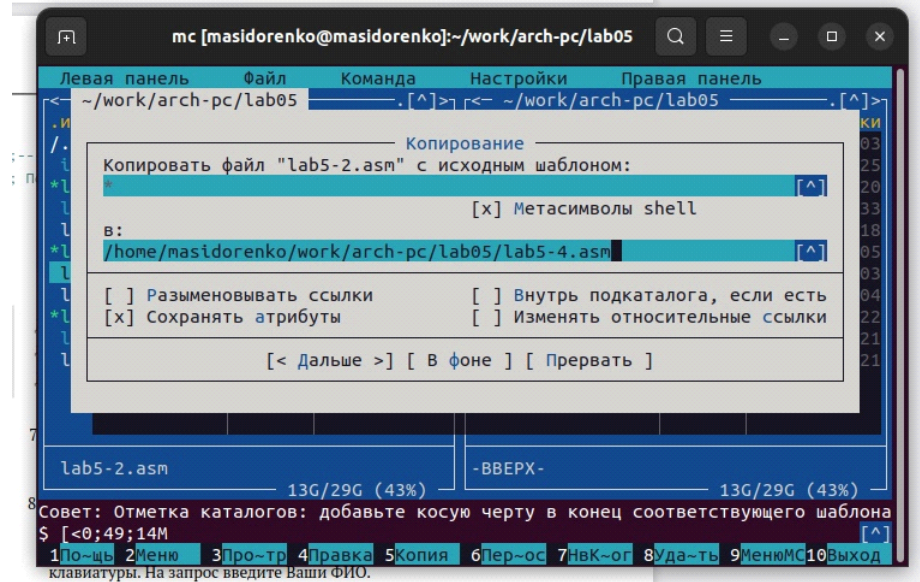
```
user@dk4n31:~$ nasm -f elf lab6-1.asm
user@dk4n31:~$ ld -m elf_i386 -o lab6-1 lab6-1.o
user@dk4n31:~$ ./lab6-1
```

A terminal window with a dark purple background and a title bar that reads "masidorenko@masidorenko: ~". The terminal shows the following commands and output:

```
masidorenko@masidorenko:~$ mc
masidorenko@masidorenko:~/work/arch-pc/lab05$ ./lab5-3
"Введите строку:"
Сидоренко
```

The cursor is positioned at the end of the input "Сидоренко".

- Создадим копию файла lab6-2.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.



- Создадим исполняемый файл и проверим его работу

mc [masidorenko@masidorenko]:~/work/arch-pc/lab05

Левая панель	Файл	Команда	Настройки	Правая панель
./	Имя	Размер	Время правки	./
in_out.asm	3942	дек 7 22:25		in_out.asm
*lab5-1	8772	дек 7 22:20		*lab5-1
lab5-1.asm	2438	дек 7 22:33		lab5-1.asm
lab5-1.o	768	дек 7 22:18		lab5-1.o
*lab5-2	9092	дек 7 23:05		*lab5-2
lab5-2.asm	1224	дек 7 23:03		lab5-2.asm
lab5-2.o	1312	дек 7 23:04		lab5-2.o
*lab5-3	8772	дек 7 23:22		*lab5-3
lab5-3.asm	2440	дек 7 23:21		lab5-3.asm
lab5-3.o	768	дек 7 23:21		lab5-3.o
lab5-4.asm	1226	дек 7 23:29		lab5-4.asm

Совет: Отметка каталогов: добавьте косую черту в конец соответствующего шаблона

```
$ nasm -f elf lab5-4.asm
```

1По-нь 2Меню 3Про-тр 4Правка 5Копия 6Пер-ос 7Нвк-ог 8Уда-ть 9МенюМС10Выход

user@dk4n31:~\$ nasm -f elf lab6-1.asm

mc [masidorenko@masidorenko]:~/work/arch-pc/lab05

Левая панель	Файл	Команда	Настройки	Правая панель
./	Имя	Размер	Время правки	./
in_out.asm	3942	дек 7 22:25		in_out.asm
*lab5-1	8772	дек 7 22:20		*lab5-1
lab5-1.asm	2438	дек 7 22:33		lab5-1.asm
lab5-1.o	768	дек 7 22:18		lab5-1.o
*lab5-2	9092	дек 7 23:05		*lab5-2
lab5-2.asm	1224	дек 7 23:03		lab5-2.asm
lab5-2.o	1312	дек 7 23:04		lab5-2.o
*lab5-3	8772	дек 7 23:22		*lab5-3
lab5-3.asm	2440	дек 7 23:21		lab5-3.asm
lab5-3.o	768	дек 7 23:21		lab5-3.o
lab5-4.asm	1226	дек 7 23:29		lab5-4.asm
lab5-4.o	1312	дек 7 23:30		lab5-4.o

Совет: Отметка каталогов: добавьте косую черту в конец соответствующего шаблона

```
$ ld -m elf_i386 -o lab5-4 lab5-4.o
```

1По-нь 2Меню 3Про-тр 4Правка 5Копия 6Пер-ос 7Нвк-ог 8Уда-ть 9МенюМС10Выход

user@dk4n31:~\$ nasm -f elf lab6-1.asm

user@dk4n31:~\$ ld -m elf_i386 -o lab6-1 lab6-1.o

user@dk4n31:~\$./lab6-1

"Введите строку:"

```
$ nasm -f elf lab5-4.asm
```

\$ ld -m elf_i386 -o lab5-4 lab5-4.o

\$./lab5-4

"Введите строку:"

Максим Сидоренко

- Вывод: после проделанной работы, я приобрел практические навыки работы в Midnight commander. Освоил инструкции языка ассемблера mov и int

Ссылка на гитхаб

https://github.com/MaximSidorenko/study_2022-2023_arh-pc