

Отчёт по лабораторной работе №5

Дисциплина: Архитектура компьютера

Кузьмина Мария Константиновна

Содержание

| | | |
|----------|--|-----------|
| 1 | Цель работы | 4 |
| 2 | Задание | 5 |
| 3 | Выполнение лабораторной работы | 6 |
| 3.1 | Работа с mc | 6 |
| 3.2 | Структура программы на языке NASM | 7 |
| 3.3 | Подключение внешнего файла in_out.asm | 10 |
| 3.4 | Выполнение заданий для самостоятельной работы. | 15 |
| 4 | Выводы | 19 |

Список иллюстраций

| | | |
|------|-------------------------|----|
| 3.1 | снимок экрана | 6 |
| 3.2 | снимок экрана | 7 |
| 3.3 | снимок экрана | 8 |
| 3.4 | снимок экрана | 9 |
| 3.5 | снимок экрана | 9 |
| 3.6 | снимок экрана | 9 |
| 3.7 | снимок экрана | 10 |
| 3.8 | снимок экрана | 10 |
| 3.9 | снимок экрана | 11 |
| 3.10 | снимок экрана | 12 |
| 3.11 | снимок экрана | 13 |
| 3.12 | снимок экрана | 13 |
| 3.13 | снимок экрана | 14 |
| 3.14 | снимок экрана | 14 |
| 3.15 | снимок экрана | 15 |
| 3.16 | снимок экрана | 16 |
| 3.17 | снимок экрана | 16 |
| 3.18 | снимок экрана | 17 |
| 3.19 | снимок экрана | 18 |
| 3.20 | снимок экрана | 18 |

1 Цель работы

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

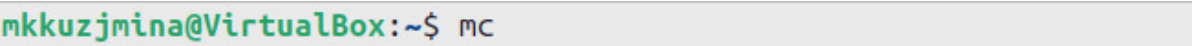
2 Задание

1. Работа с mc
2. Структура программы на языке NASM
3. Подключение внешнего файла in_out.asm
4. Выполнение заданий для самостоятельной работы.

3 Выполнение лабораторной работы

3.1 Работа с mc

Открываем Midnight Commander с помощью mc (рис. 3.1):



```
mkkuzjmina@VirtualBox:~$ mc
```

Рис. 3.1: снимок экрана

С помощью функциональной клавиши F7 создаем папку lab05 (рис. 3.2):

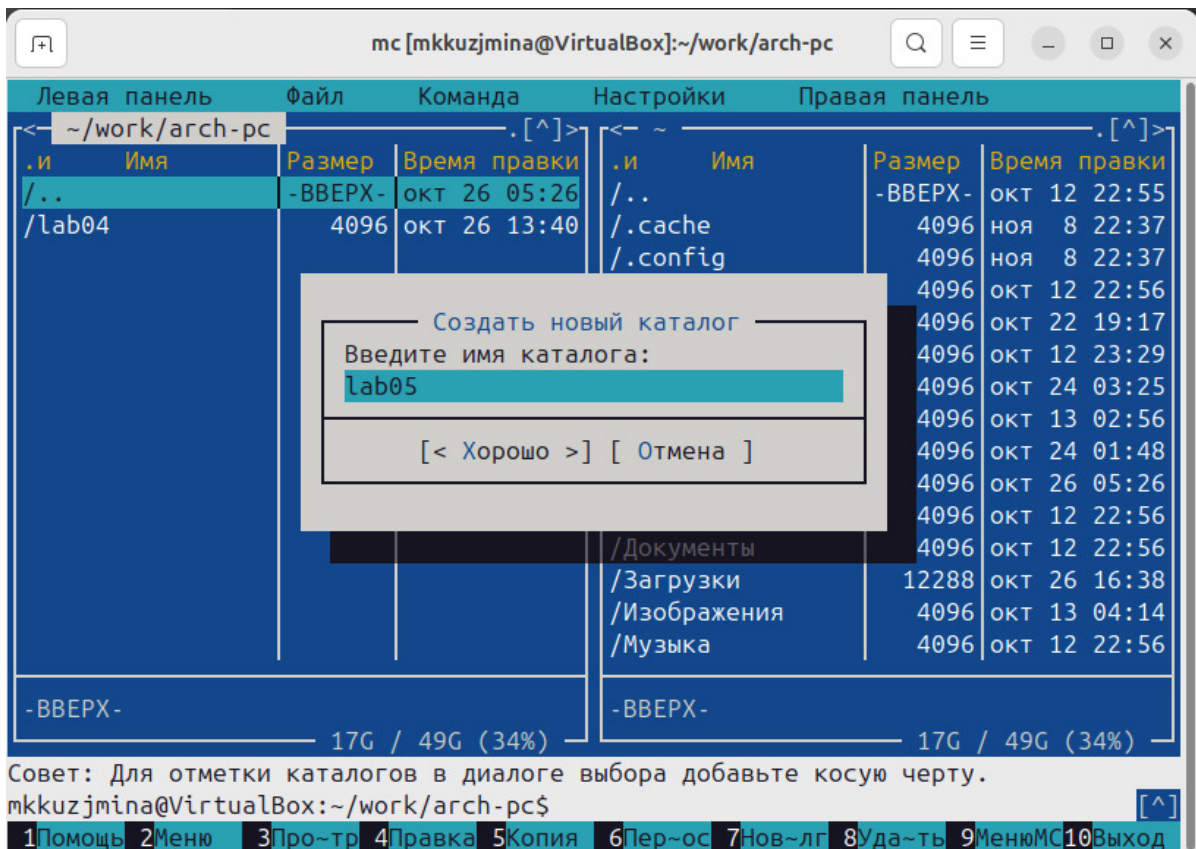
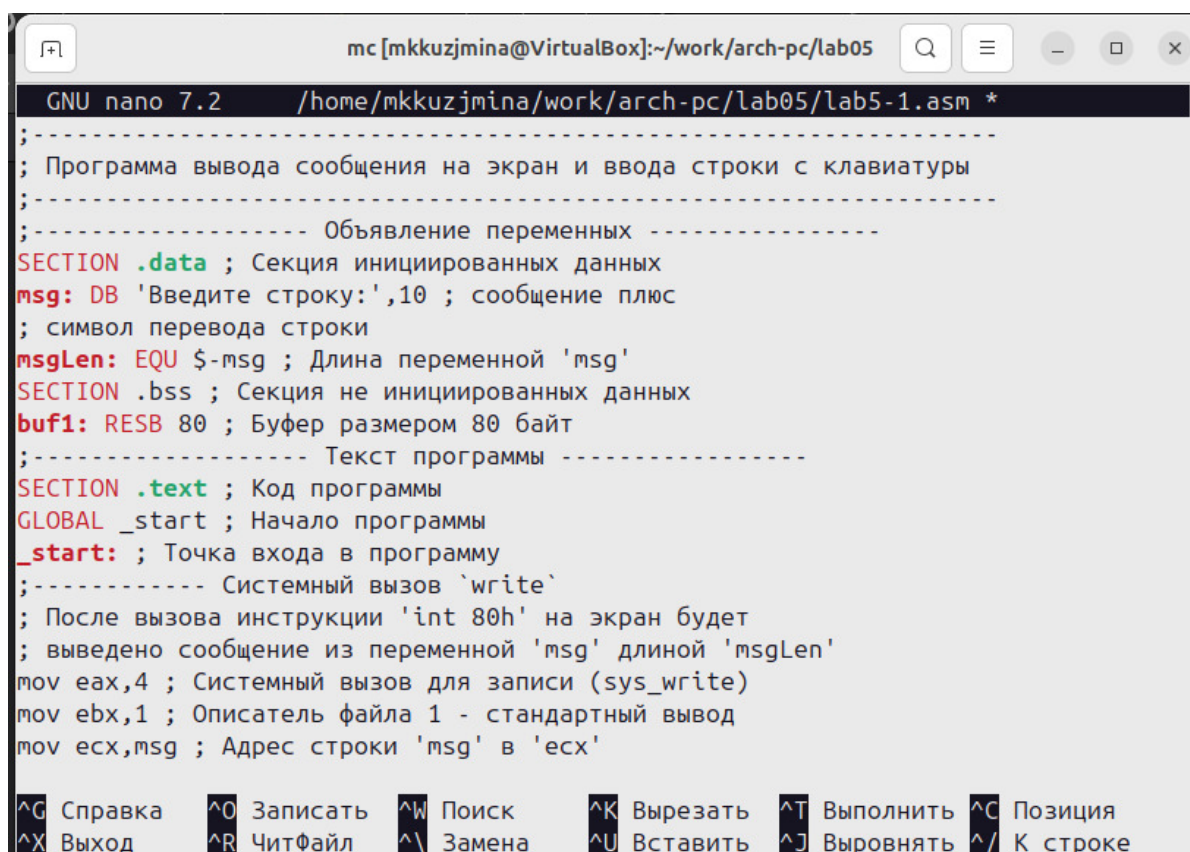


Рис. 3.2: снимок экрана

Переходим в созданный каталог, прописываем команду touch и создаем файл lab5-1.asm

3.2 Структура программы на языке NASM

С помощью функциональной клавиши F4 открываем файл lab5-1.asm для редактирования во встроенном редакторе nano, вводим текст, сохраняем изменения и закрываем файл (рис. 3.3):

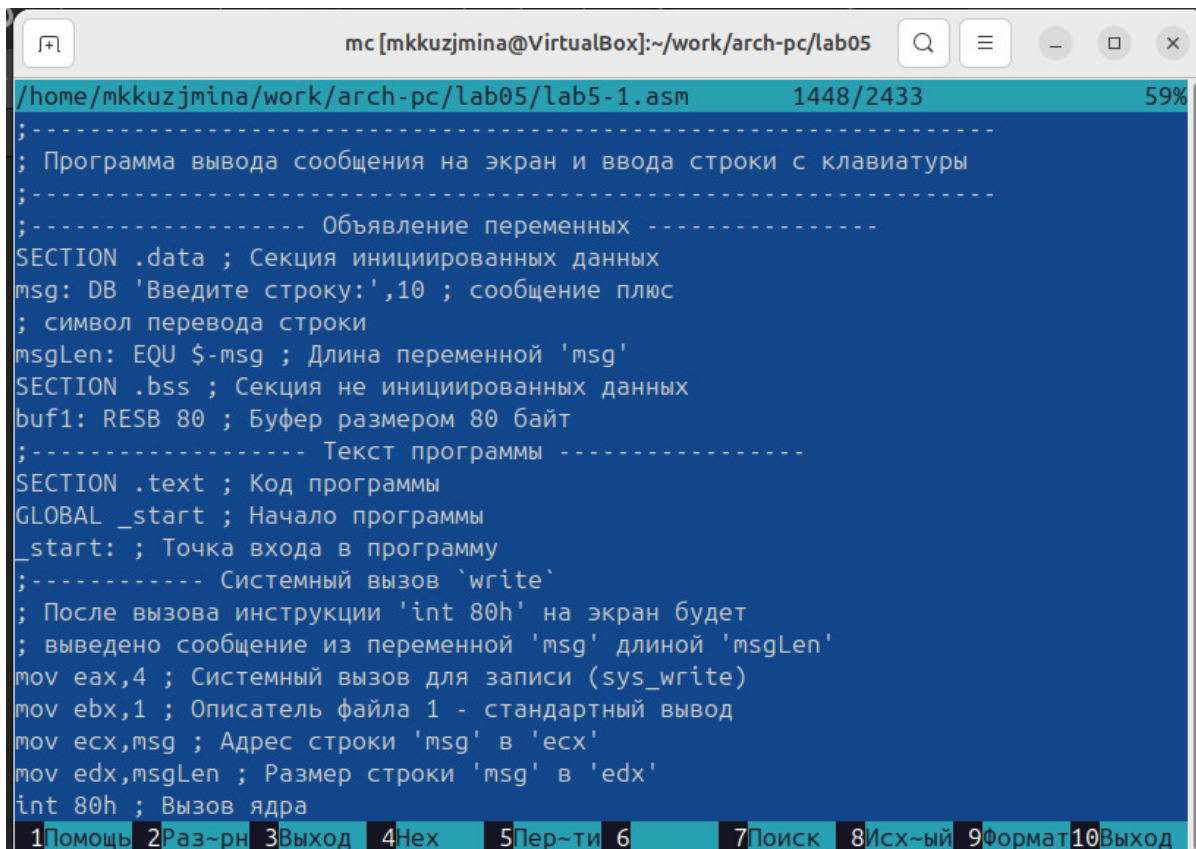


```
mc [mkkuzjmina@VirtualBox]:~/work/arch-pc/lab05
GNU nano 7.2 /home/mkkuzjmina/work/arch-pc/lab05/lab5-1.asm *
;-----
; Программа вывода сообщения на экран и ввода строки с клавиатуры
;-----
;----- Объявление переменных -----
SECTION .data ; Секция инициированных данных
msg: DB 'Введите строку:',10 ; сообщение плюс
; символ перевода строки
msgLen: EQU $-msg ; Длина переменной 'msg'
SECTION .bss ; Секция не инициированных 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'

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

Рис. 3.3: снимок экрана

С помощью функциональной клавиши F3 открываем файл lab5-1.asm для просмотра.(рис. 3.4):



```
mc [mkkuzjmina@VirtualBox]:~/work/arch-pc/lab05
/home/mkkuzjmina/work/arch-pc/lab05/lab5-1.asm 1448/2433 59%
;-----
; Программа вывода сообщения на экран и ввода строки с клавиатуры
;-----
;----- Объявление переменных -----
SECTION .data ; Секция инициированных данных
msg: DB 'Введите строку:',10 ; сообщение плюс
; символ перевода строки
msgLen: EQU $-msg ; Длина переменной 'msg'
SECTION .bss ; Секция не инициированных 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
1Помощь 2Раз~рн 3Выход 4Нех 5Пер~ти 6 7Поиск 8Исх~ый 9Формат10Выход
```

Рис. 3.4: снимок экрана

Транслируем текст программы lab5-1.asm в объектный файл. Выполняем компо- новку объектного файла.(рис. 3.5):

```
mkkuzjmina@VirtualBox:~/work/arch-pc/lab05$ nasm -f elf lab5-1.asm
```

Рис. 3.5: снимок экрана

(рис. 3.6):

```
mkkuzjmina@VirtualBox:~/work/arch-pc/lab05$ ld -m elf_i386 -o lab5-1 lab5-1.o
```

Рис. 3.6: снимок экрана

Создался файл lab5-1.0 (рис. 3.7):

| Левая панель | | | Файл | Команда | Настройки | Правая панель | | |
|-------------------------|--|--|---------|-------------|--------------|-----------------|--|--|
| < ~ /work/arch-pc/lab05 | | | | | | < ~ | | |
| .и | | | Имя | Размер | Время правки | .и | | |
| /.. | | | -ВВЕРХ- | ноя 8 23:09 | | /.. | | |
| lab5-1.asm | | | 2433 | ноя 8 23:20 | | /.cache | | |
| lab5-1.o | | | 752 | ноя 8 23:24 | | /.config | | |
| | | | | | | /.local | | |
| | | | | | | /.pandoc | | |
| | | | | | | /.ssh | | |
| | | | | | | /.texlive2023 | | |
| | | | | | | /.texlive2024 | | |
| | | | | | | /snap | | |
| | | | | | | /work | | |
| | | | | | | /Видео | | |
| | | | | | | /Документы | | |
| | | | | | | /Загрузки | | |
| | | | | | | /Изображения | | |
| | | | | | | /Музыка | | |
| lab5-1.o | | | | | | -ВВЕРХ- | | |
| 17G / 49G (34%) | | | | | | 17G / 49G (34%) | | |

Рис. 3.7: снимок экрана

Запускаем исполняемый файл и вводим ФИО

(рис. 3.8):

```

mkkuzjmina@VirtualBox:~/work/arch-pc/lab05$ ./lab5-1
Введите строку:
Кузьмина Мария Константиновна

```

Рис. 3.8: снимок экрана

3.3 Подключение внешнего файла in_out.asm

Скачиваем файл in_out.asm. Копируем файл in_out.asm в каталог с файлом lab5-1.asm с помощью функциональной клавиши F5(рис. 3.9):

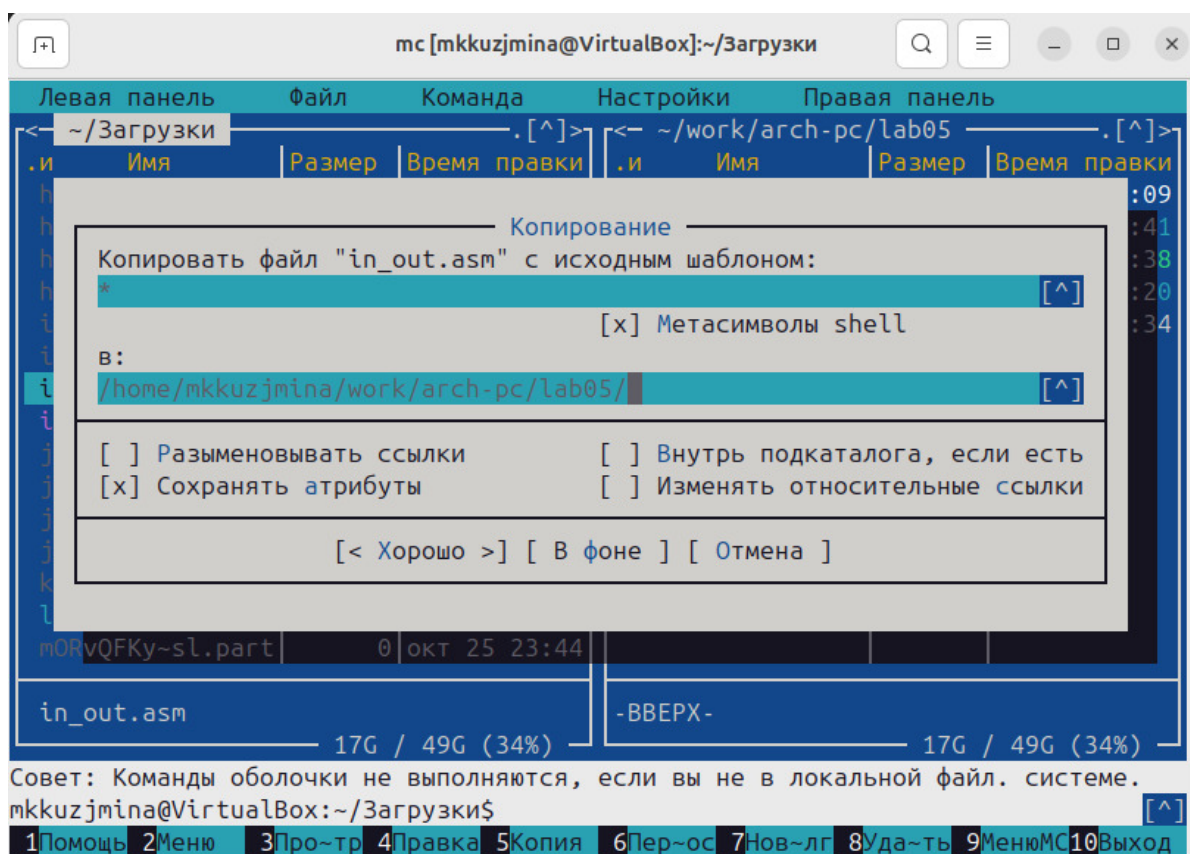


Рис. 3.9: снимок экрана

С помощью функциональной клавиши F5 создаем копию файла lab5-1.asm с именем lab5-2.asm.(рис. 3.10):

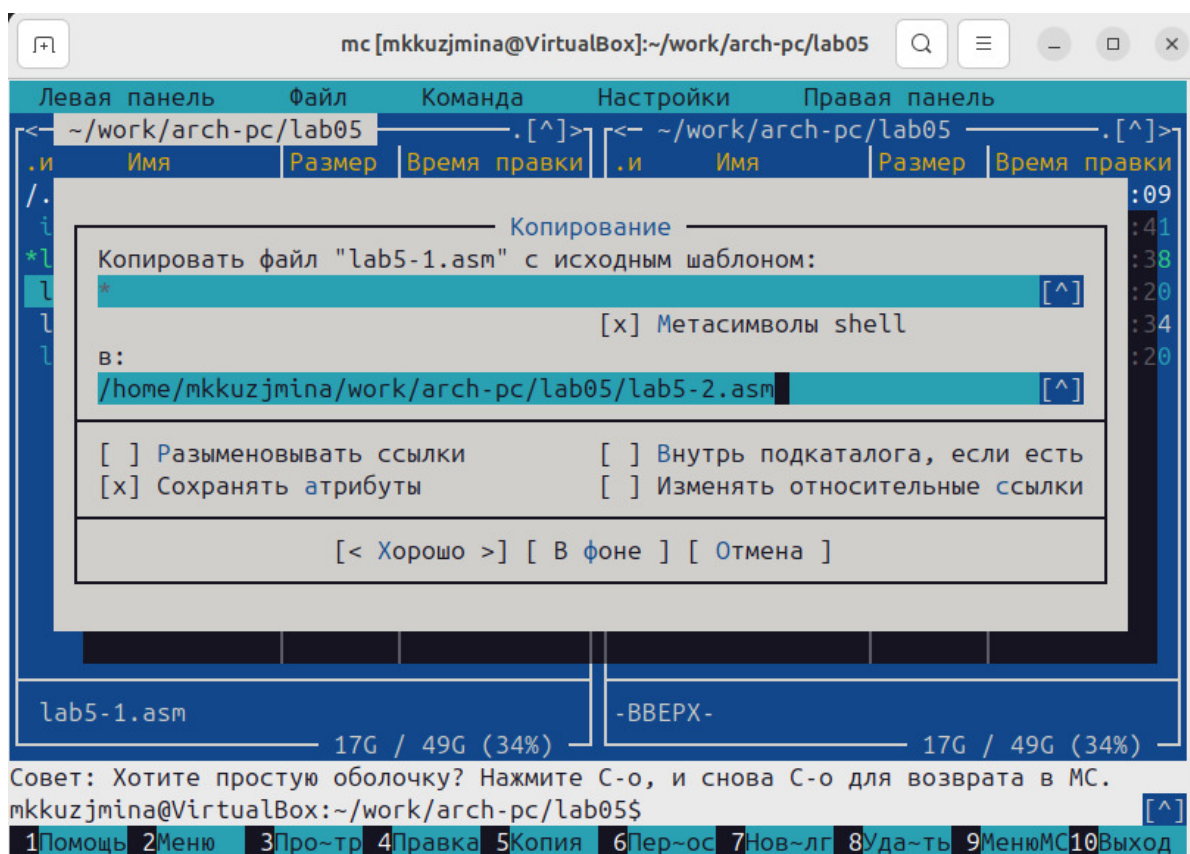
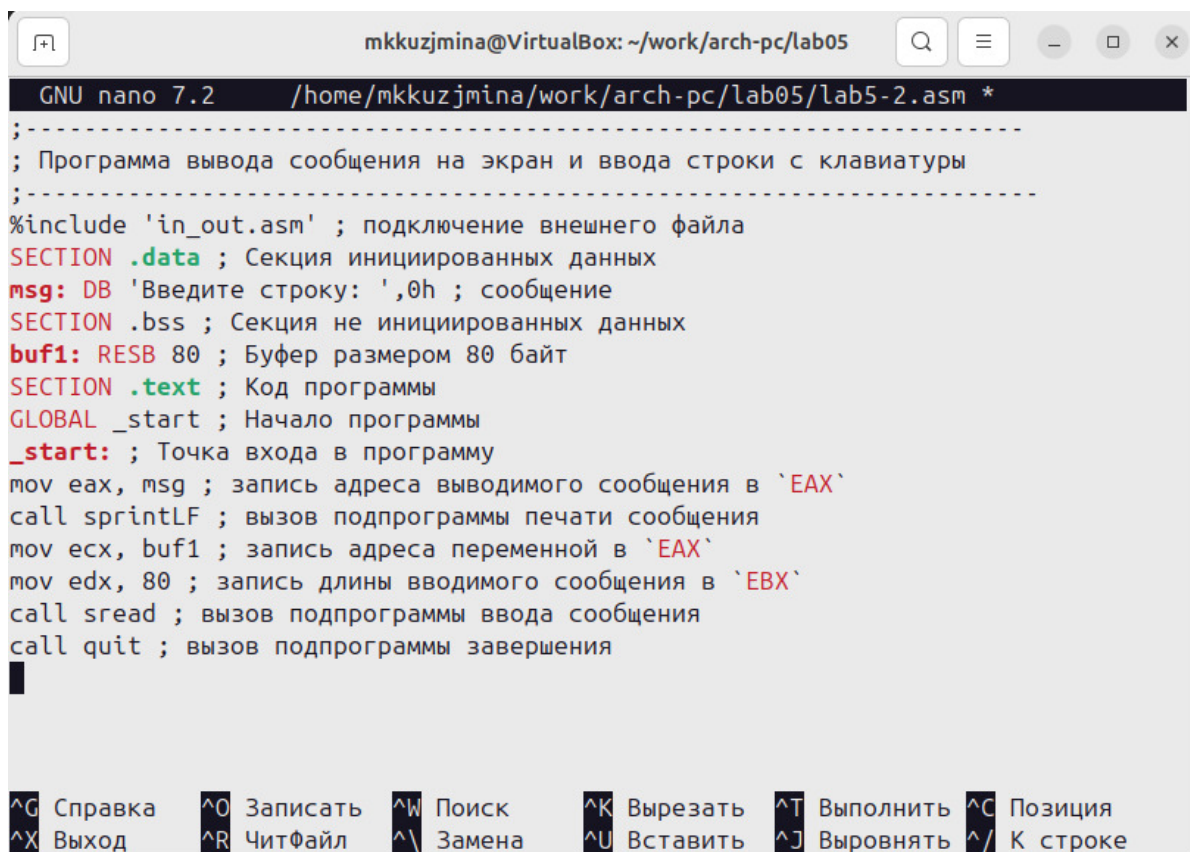


Рис. 3.10: снимок экрана

Исправляем текст программы в файле lab5-2.asm с использованием подпрограмм из внешнего файла in_out.asm (рис. 3.11):

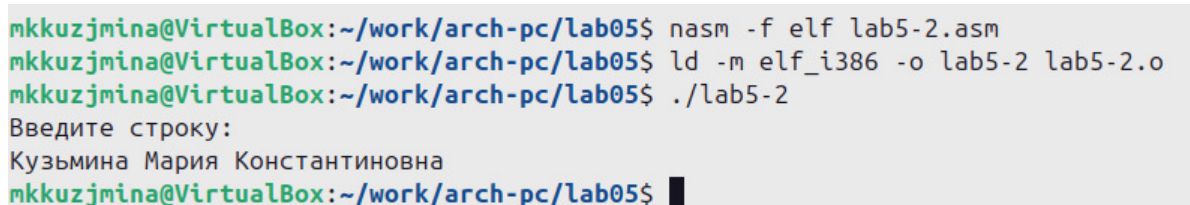


```
GNU nano 7.2 /home/mkkuzjmina/work/arch-pc/lab05/lab5-2.asm *
;-----
; Программа вывода сообщения на экран и ввода строки с клавиатуры
;-----
#include 'in_out.asm' ; подключение внешнего файла
SECTION .data ; Секция инициированных данных
msg: DB 'Введите строку: ',0h ; сообщение
SECTION .bss ; Секция 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 ; 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

```

Рис. 3.11: снимок экрана

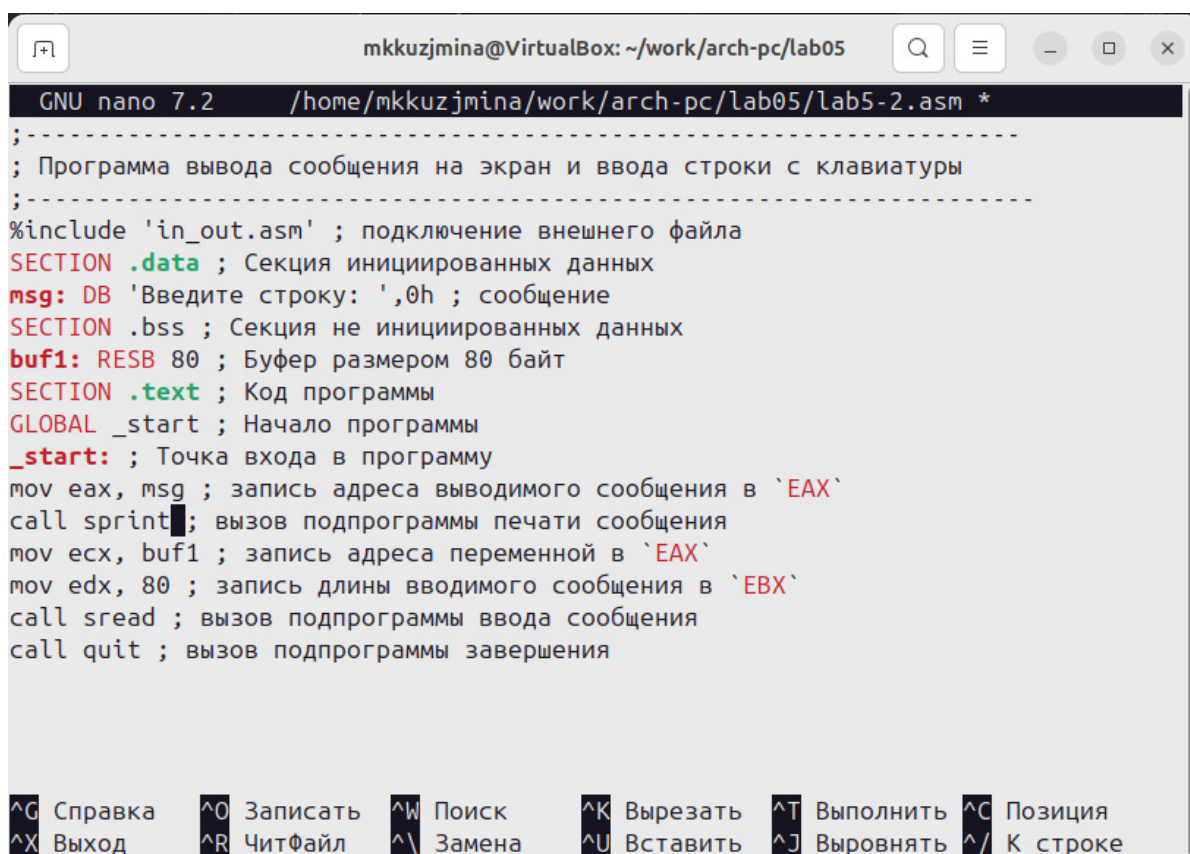
Транслируем текст программы lab5-2.asm в объектный файл. Выполняем компоновку объектного файла. Создался файл lab5-2.o. Запускаем исполняемый файл и вводим ФИО. (рис. 3.12):



```
mkkuzjmina@VirtualBox:~/work/arch-pc/lab05$ nasm -f elf lab5-2.asm
mkkuzjmina@VirtualBox:~/work/arch-pc/lab05$ ld -m elf_i386 -o lab5-2 lab5-2.o
mkkuzjmina@VirtualBox:~/work/arch-pc/lab05$ ./lab5-2
Введите строку:
Кузьмина Мария Константиновна
mkkuzjmina@VirtualBox:~/work/arch-pc/lab05$
```

Рис. 3.12: снимок экрана

Открываем файл lab5-2.asm для редактирования в nano функциональной клавишей F4. Изменяем sprintf на sprint. Сохраняем изменения. (рис. 3.13):

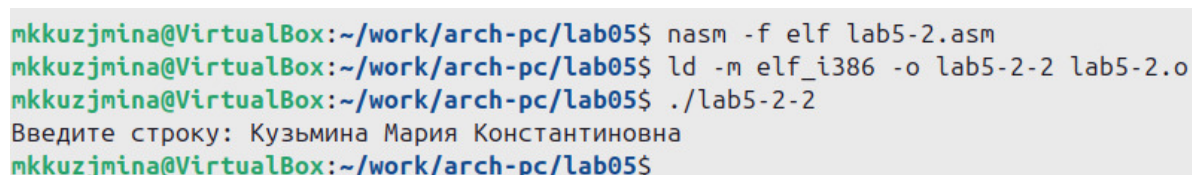


```
GNU nano 7.2 /home/mkkuzjmina/work/arch-pc/lab05/lab5-2.asm *
;-----
; Программа вывода сообщения на экран и ввода строки с клавиатуры
;-----
#include 'in_out.asm' ; подключение внешнего файла
SECTION .data ; Секция инициированных данных
msg: DB 'Введите строку: ',0h ; сообщение
SECTION .bss ; Секция 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 ; 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     ^W Поиск       ^K Вырезать    ^T Выполнить   ^C Позиция
^X Выход        ^R ЧитФайл     ^\ Замена      ^U Вставить    ^J Выровнять   ^/ К строке
```

Рис. 3.13: снимок экрана

Транслируем файл, выполняем компоновку созданного объектного файла, за-
пускаем исполняемый файл. (рис. 3.14):



```
mkkuzjmina@VirtualBox:~/work/arch-pc/lab05$ nasm -f elf lab5-2.asm
mkkuzjmina@VirtualBox:~/work/arch-pc/lab05$ ld -m elf_i386 -o lab5-2-2 lab5-2.o
mkkuzjmina@VirtualBox:~/work/arch-pc/lab05$ ./lab5-2-2
Введите строку: Кузьмина Мария Константиновна
mkkuzjmina@VirtualBox:~/work/arch-pc/lab05$
```

Рис. 3.14: снимок экрана

После запуска второго файла мы видим, что разница заключается в том, что
первый файл запрашивает ввод с новой строки, тогда как второй — без переноса
на новую строку. `sprintf` добавляет символ новой строки после ввода, а `sprint`
этого не делает.

3.4 Выполнение заданий для самостоятельной работы.

Создаем копию файла lab5-1.asm с именем lab5-1-1.asm с помощью функциональной клавиши F5 (рис. 3.15):

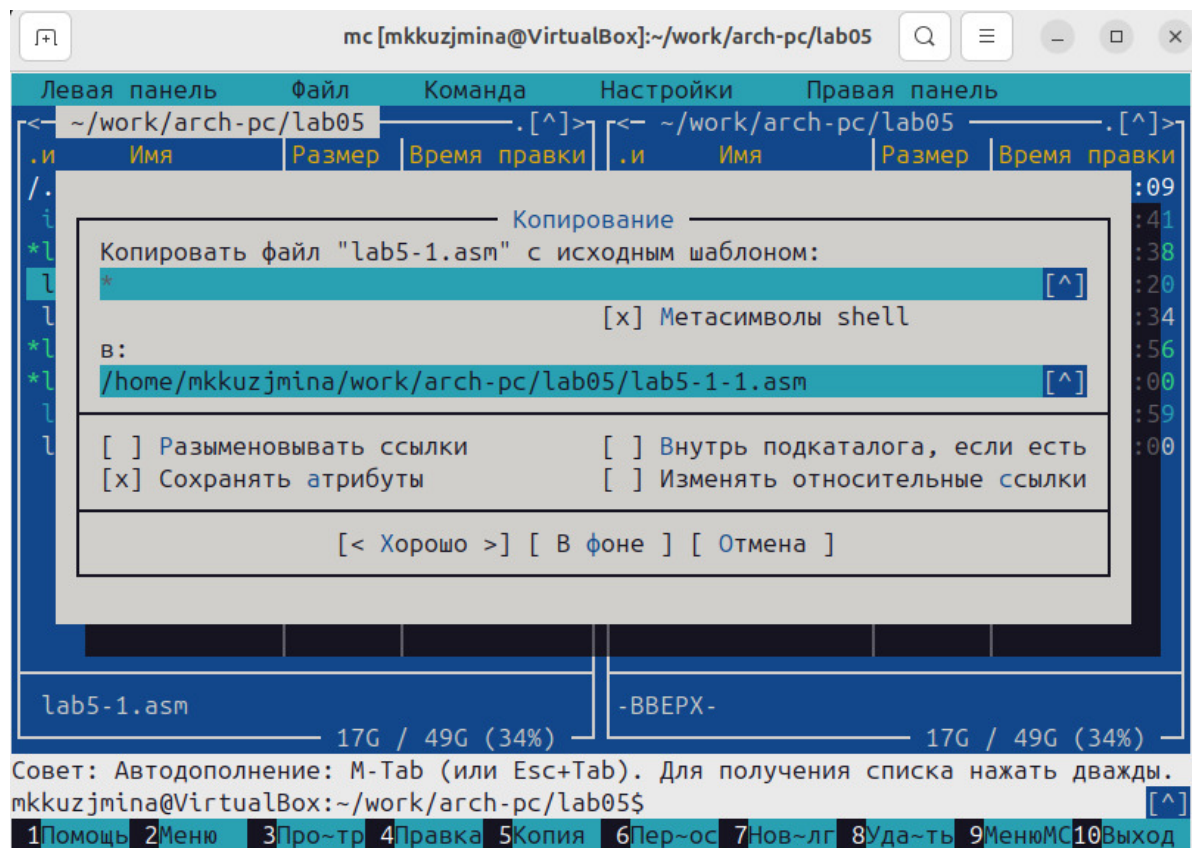
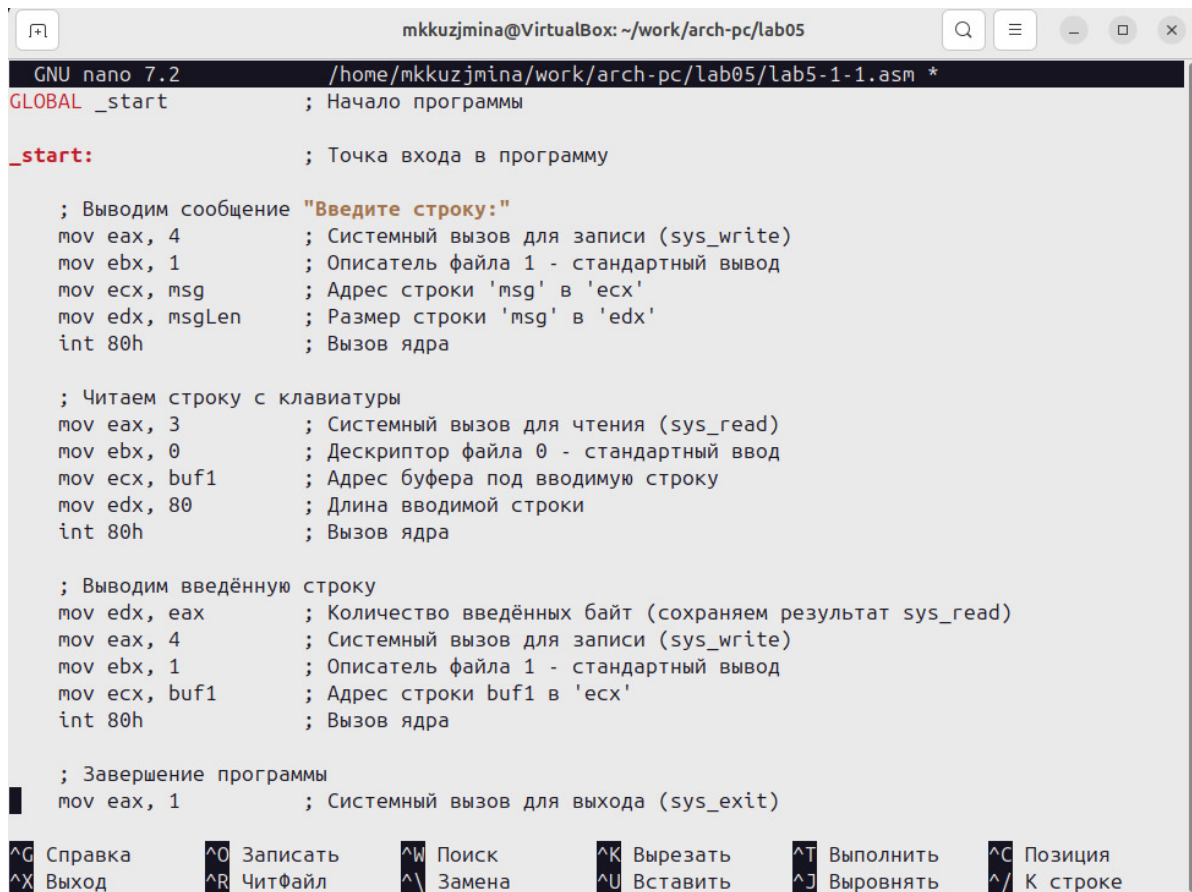


Рис. 3.15: снимок экрана

С помощью функциональной клавиши F4 открываем созданный файл для редактирования. Изменяем программу, чтобы она выводила вводимую пользователем строку. (рис. 3.16):



```
GNU nano 7.2 /home/mkkuzjmina/work/arch-pc/lab05/lab5-1-1.asm *
GLOBAL _start ; Начало программы

_start: ; Точка входа в программу

; Выводим сообщение "Введите строку:"
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

; Читаем строку с клавиатуры
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

; Выводим введенную строку
mov edx, eax ; Количество введенных байт (сохраняем результат sys_read)
mov eax, 4 ; Системный вызов для записи (sys_write)
mov ebx, 1 ; Описатель файла 1 - стандартный вывод
mov ecx, buf1 ; Адрес строки buf1 в 'ecx'
int 80h ; Вызов ядра

; Завершение программы
mov eax, 1 ; Системный вызов для выхода (sys_exit)
```

Рис. 3.16: снимок экрана

Транслируем текст программы lab5-1-1.asm в объектный файл. Выполняем компоновку объектного файла. Создался файл lab5-1-1.o. Запускаем исполняемый файл и вводим ФИО. (рис. 3.17):

```
mkkuzjmina@VirtualBox:~/work/arch-pc/lab05$ nasm -f elf lab5-1-1.asm
mkkuzjmina@VirtualBox:~/work/arch-pc/lab05$ ld -m elf_i386 -o lab5-1-1 lab5-1-1.o
mkkuzjmina@VirtualBox:~/work/arch-pc/lab05$ ./lab5-1-1
Введите строку:
Кузьмина Мария Константиновна
Кузьмина Мария Константиновна
mkkuzjmina@VirtualBox:~/work/arch-pc/lab05$
```

Рис. 3.17: снимок экрана

Создаем копию файла lab5-2.asm (рис. 3.18):

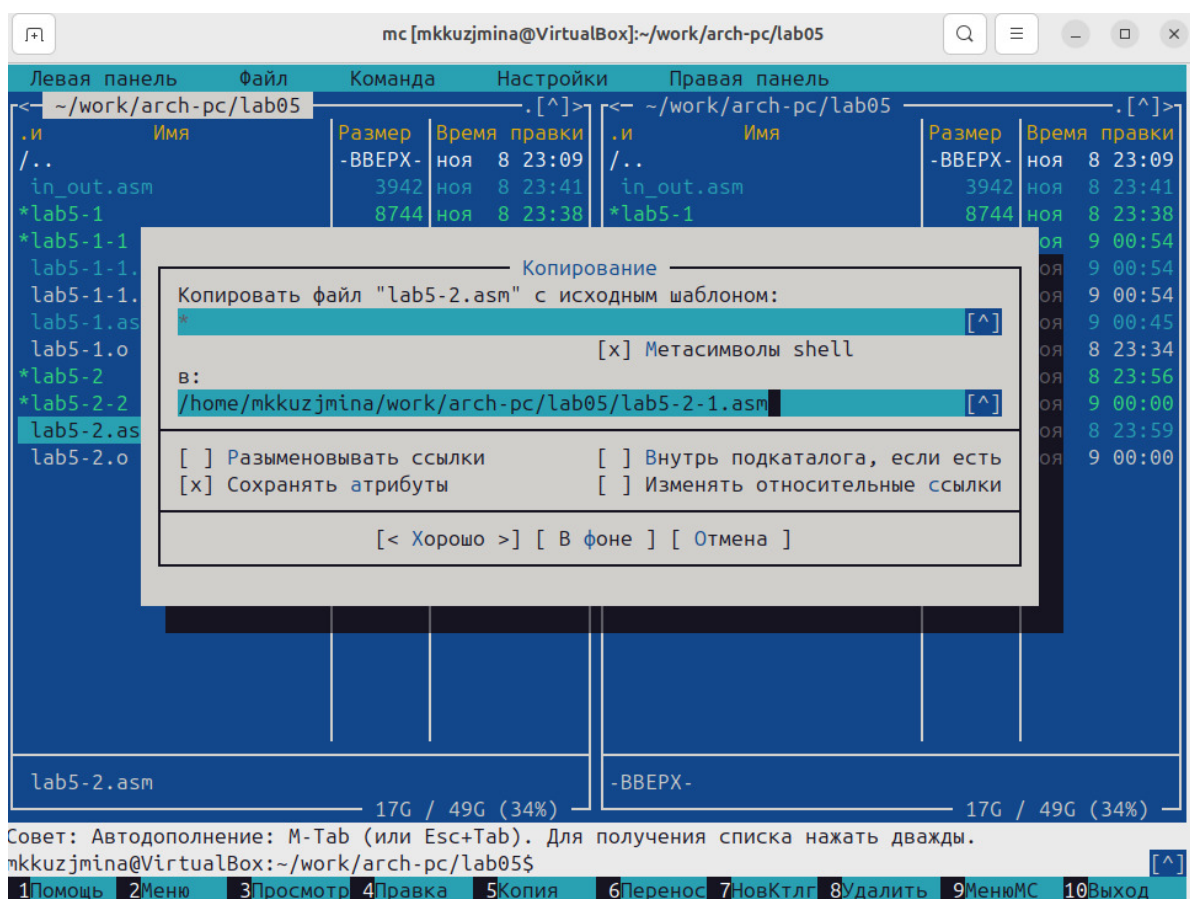
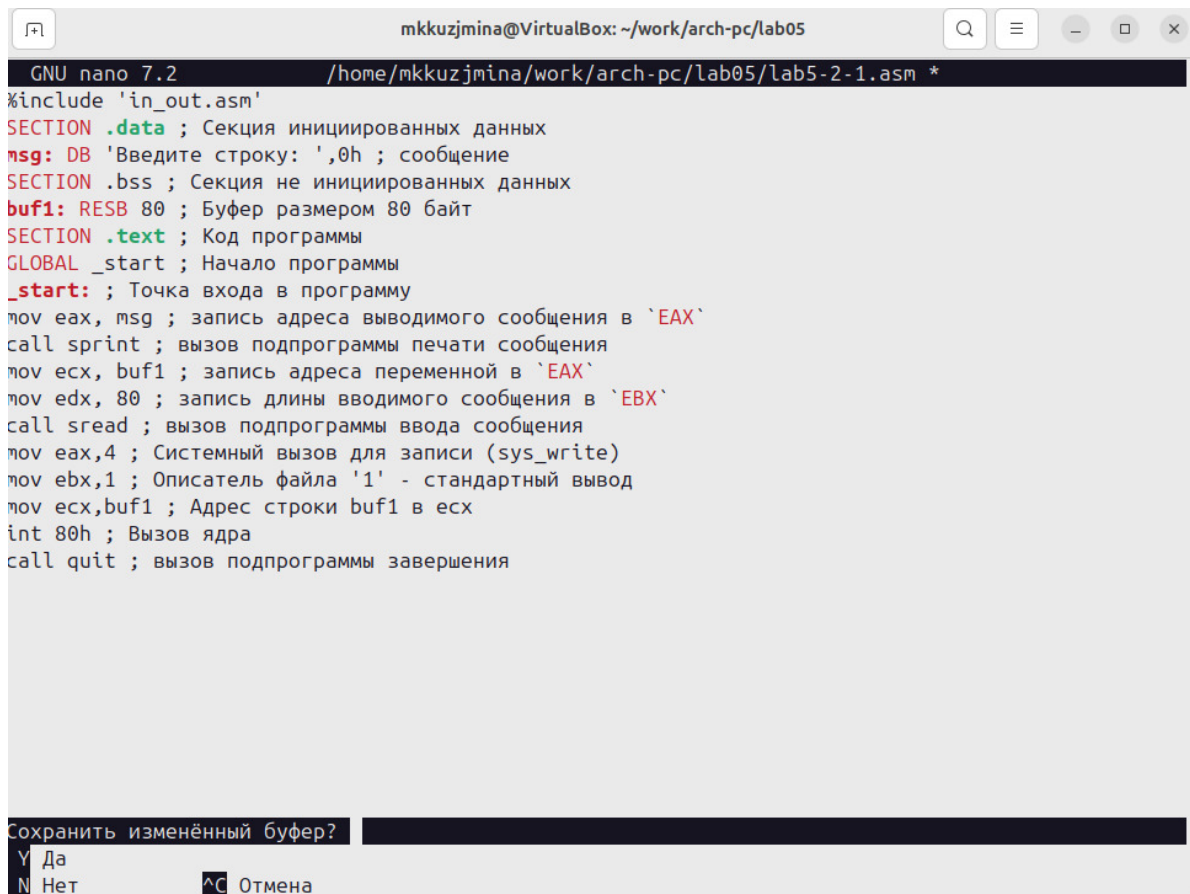


Рис. 3.18: снимок экрана

С помощью функциональной клавиши F4 открываем созданный файл для редактирования. Изменяем программу, чтобы она выводила вводимую пользователем строку (рис. 3.19):



```
GNU nano 7.2 /home/mkkuzjmina/work/arch-pc/lab05/lab5-2-1.asm *
%include 'in_out.asm'
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 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
mov eax,4 ; Системный вызов для записи (sys_write)
mov ebx,1 ; Описатель файла '1' - стандартный вывод
mov ecx,buf1 ; Адрес строки buf1 в ecx
int 80h ; Вызов ядра
call quit ; вызов подпрограммы завершения

Сохранить изменённый буфер?
Y Да
N Нет ^C Отмена
```

Рис. 3.19: снимок экрана

Создаем объектный файл lab5-2-1.o, запускаем полученный исполняемый файл, вводим ФИО (рис. 3.20):

```
mkkuzjmina@VirtualBox:~/work/arch-pc/lab05$ nasm -f elf lab5-2-1.asm
mkkuzjmina@VirtualBox:~/work/arch-pc/lab05$ ld -m elf_i386 -o lab5-2-1 lab5-2-1.o
mkkuzjmina@VirtualBox:~/work/arch-pc/lab05$ ./lab5-2-1
Введите строку: Кузьмина Мария Константиновна
Кузьмина Мария Константиновна
mkkuzjmina@VirtualBox:~/work/arch-pc/lab05$
```

Рис. 3.20: снимок экрана

4 Выводы

При выполнении лабораторной работы были приобретены практические навыки работы в Midnight Commander, а также освоены инструкции языка ассемблера `mov` и `int`.