

Лабораторная работа №5

Основы работы с Midnight Commander (mc). Структура программы на языке ассемблера NASM. Системные вызовы в ОС GNU Linux

Мочалкина Софья Васильевна

Содержание

Цель работы	1
Задание	1
Теоретическое введение	2
Выполнение лабораторной работы	2
Выводы.....	11
Список литературы.....	11

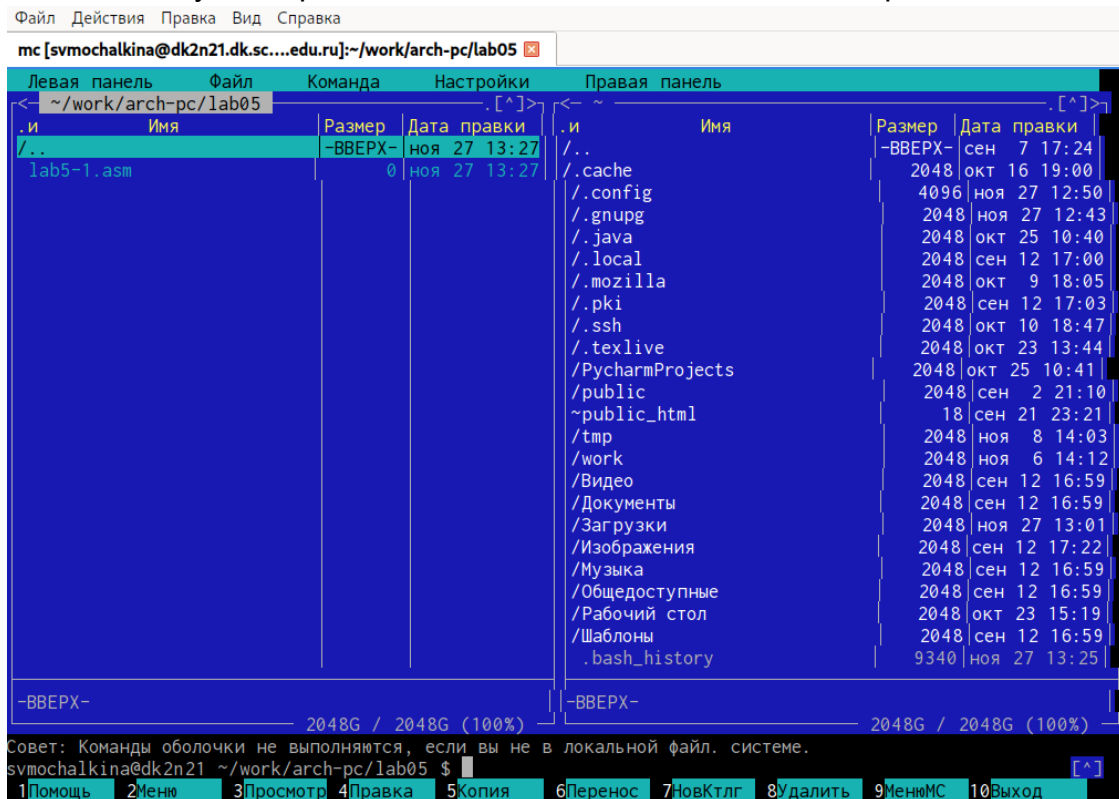
Цель работы

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

Задание

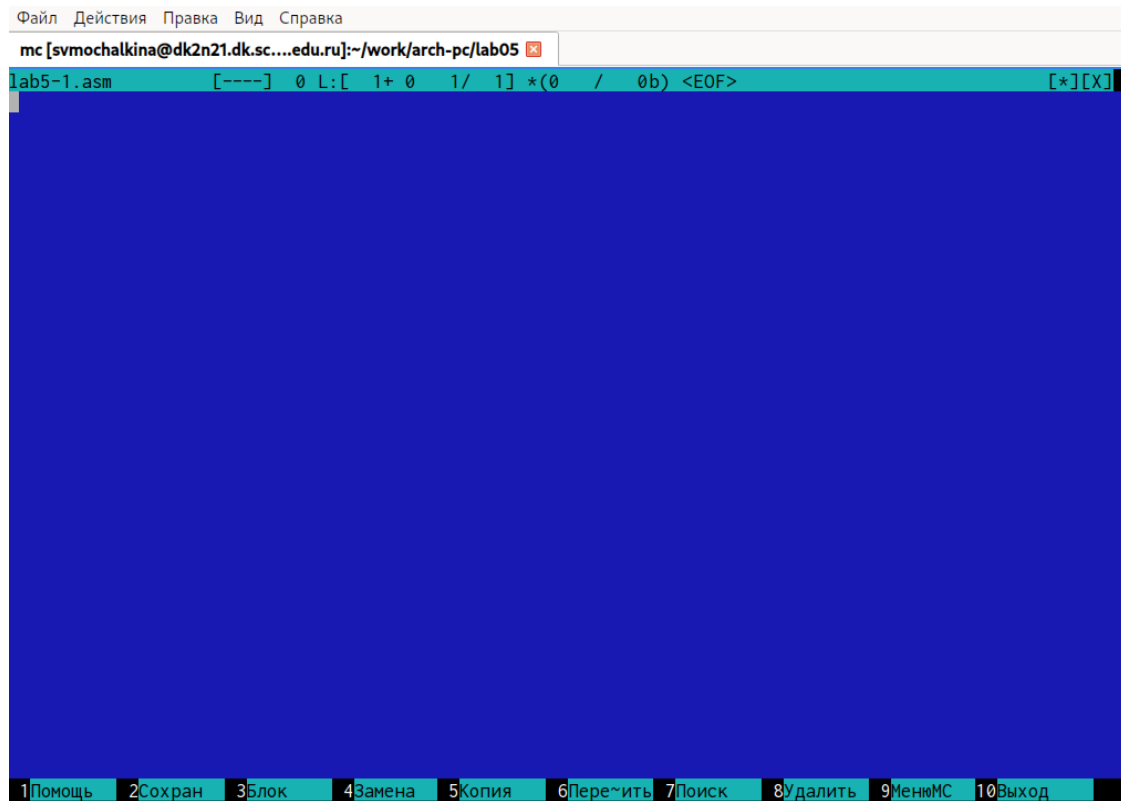
- 1.Открыть Midnight Commander
- 2.Создать папку lab05 и внутри нее создать файл lab5-1.asm
- 3.Открыть файл lab5-1.asm, ввести информацию из листинга 5.1 и сохранить изменения
- 4.Убедится что файл содержит информацию
- 5.Оттранслировать текст файла lab5-1.asm, выполнить компоновку объектного файла
- 6.Запустить файл
- 7.Скачать и скопировать файл in_out.asm с помощью клавиши f5
- 8.С помощью клавиши f6 скопировать файл lab5-1.asm с именем lab5-2.asm
- 9.Исправить файл lab5-2.asm в соответствии с листингом 5.2
- 10.В файле lab5-2.asm заменить подпрограмму sprintLF на sprint

3. Пользуясь строкой ввода и командой touch создаю файл lab5-1.asm (рис.3)



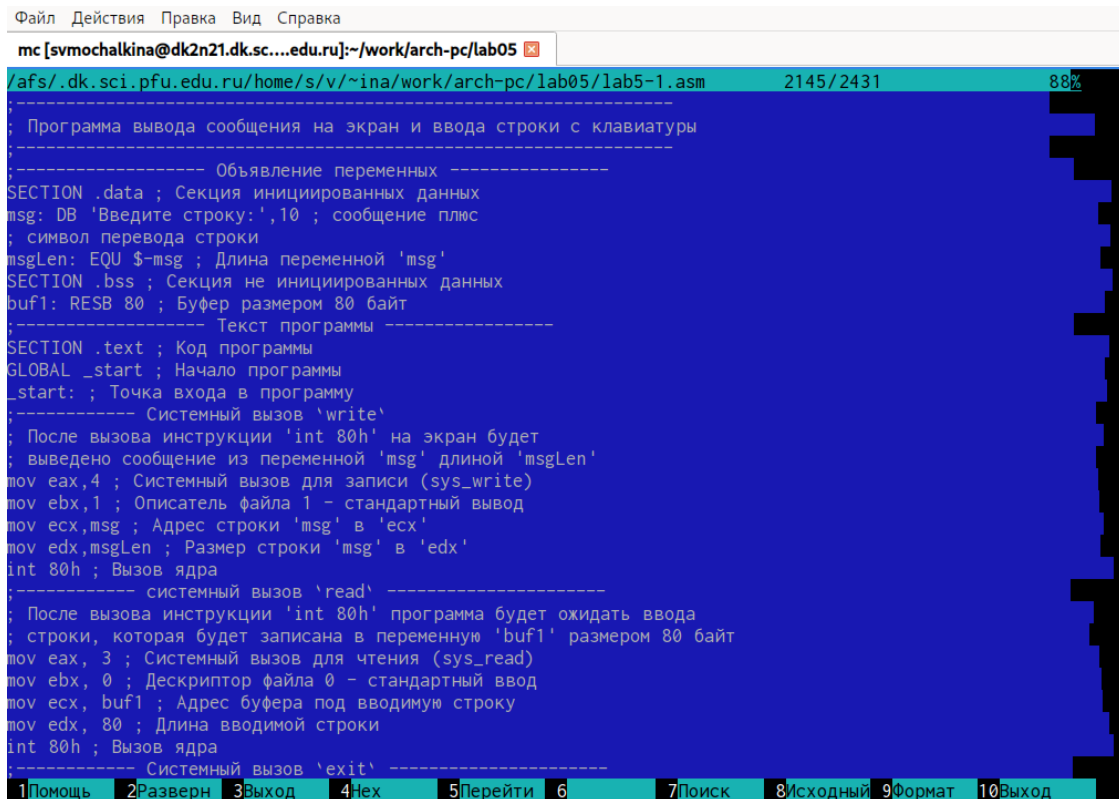
создание файла lab5-1.asm(рис. 3)

4. С помощью функциональной клавиши F4 открываю файл lab5-1.asm для редактирования во встроенном редакторе. (рис. 4).



Новый файл lab5-1 (рис.4)

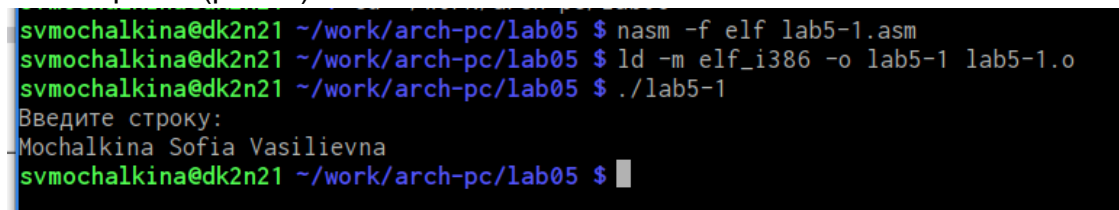
5. Ввожу текст программы из листинга (рис. 5). С помощью функциональной клавиши F3 открываю файл lab5-1.asm для просмотра. Убеждаюсь, что файл содержит текст программы.



```
Файл Действия Правка Вид Справка
mc [svmochalkina@dk2n21.dk.sc....edu.ru]:~/work/arch-pc/lab05
/afs/.dk.sci.pfu.edu.ru/home/s/v/~ina/work/arch-pc/lab05/lab5-1.asm 2145/2431 88%
; 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
;----- 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
;----- Системный вызов 'exit' -----
1Помощь 2Разверн 3Выход 4Нех 5Перейти 6 7Поиск 8Исходный 9Формат 10Выход
```

Введённый текст(рис. 5)

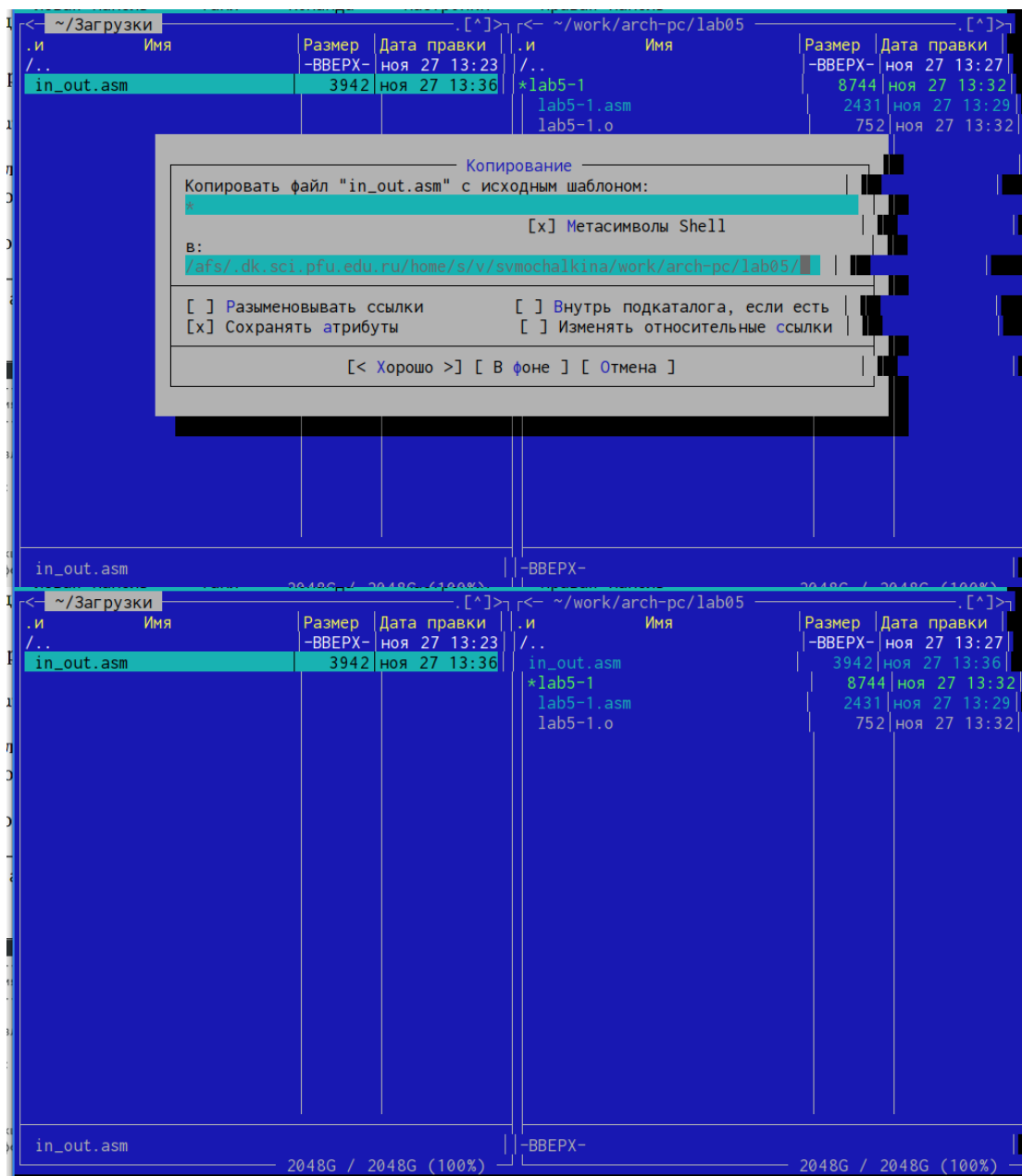
6. Транспирую текст программы lab5-1.asm в объектный файл. Выполняю компо- новку объектного файла и запускаю получившийся исполняемый файл. (рис. 6)



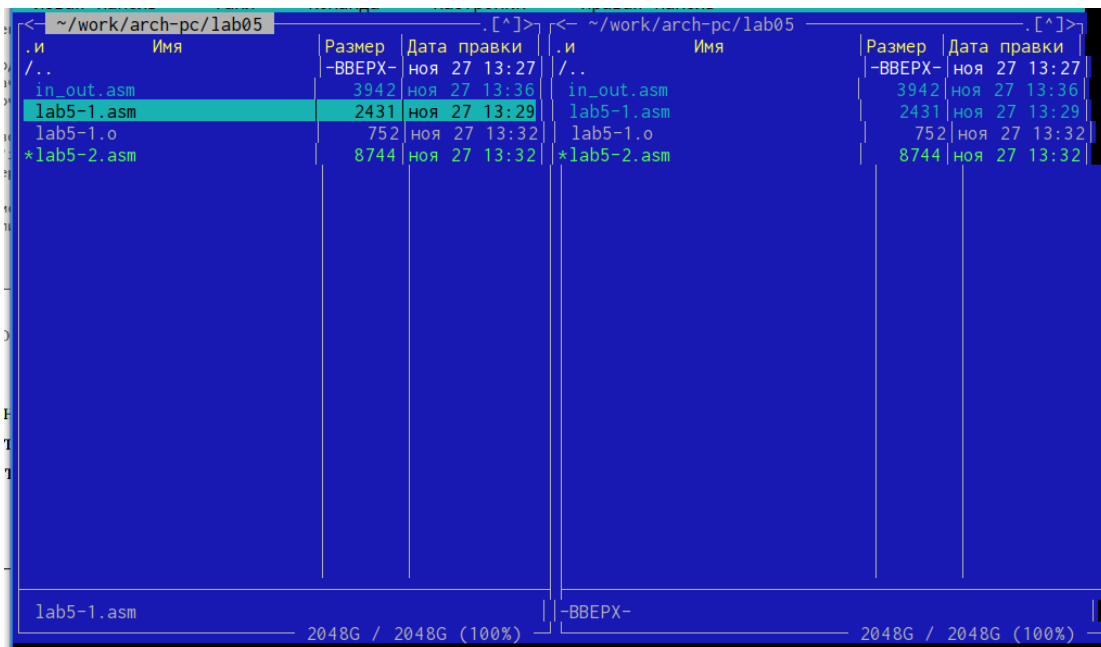
```
svmochalkina@dk2n21 ~/work/arch-pc/lab05 $ nasm -f elf lab5-1.asm
svmochalkina@dk2n21 ~/work/arch-pc/lab05 $ ld -m elf_i386 -o lab5-1 lab5-1.o
svmochalkina@dk2n21 ~/work/arch-pc/lab05 $ ./lab5-1
Введите строку:
Mochalkina Sofia Vasilievna
svmochalkina@dk2n21 ~/work/arch-pc/lab05 $
```

Выполнение файла lab5-1 (рис. 6)

7. Скачиваю файл in_out.asm со страницы курса в ТУИС. Копирую файл in_out.asm в каталог с файлом lab5-1.asm с помощью функциональной клавиши F5 (рис. 7).

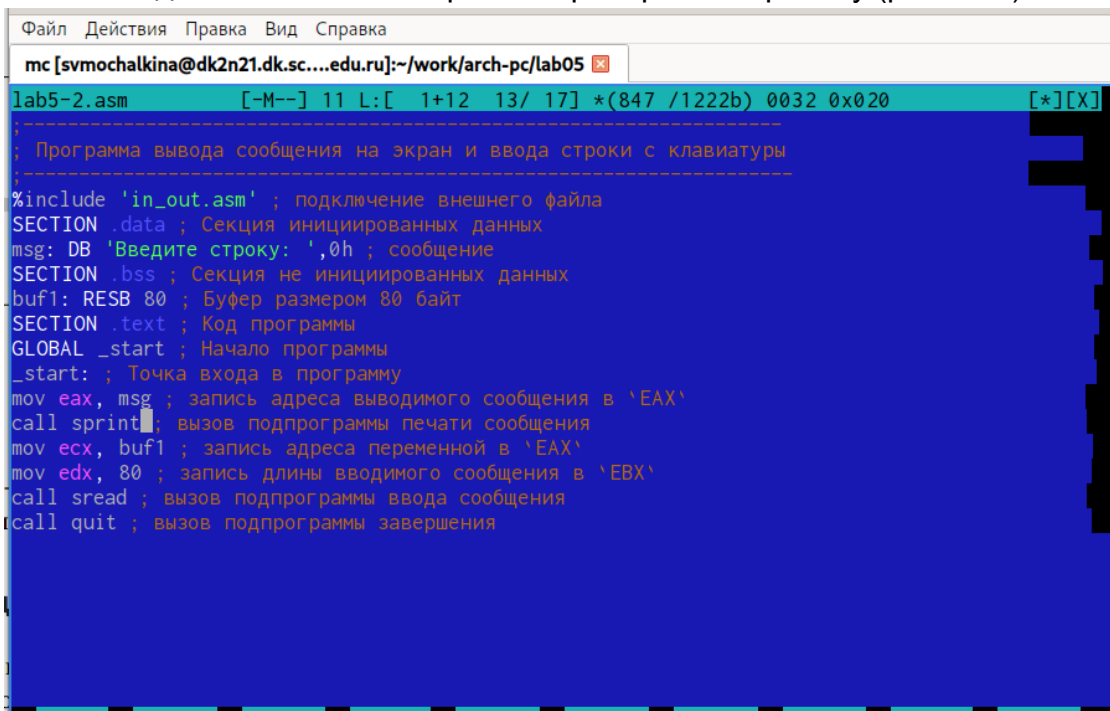


8. С помощью функциональной клавиши F6 создаю копию файла lab5-1.asm с именем lab5-2.asm. (рис. 8)



создаём *lab5-2*

9. Исправляю текст программы в файле *lab5-2.asm* с использованием подпрограмм из внешнего файла *in_out.asm* в соответствии с листингом 5.2. Создаю исполняемый файл и проверяю его работу. (рис 9-10)



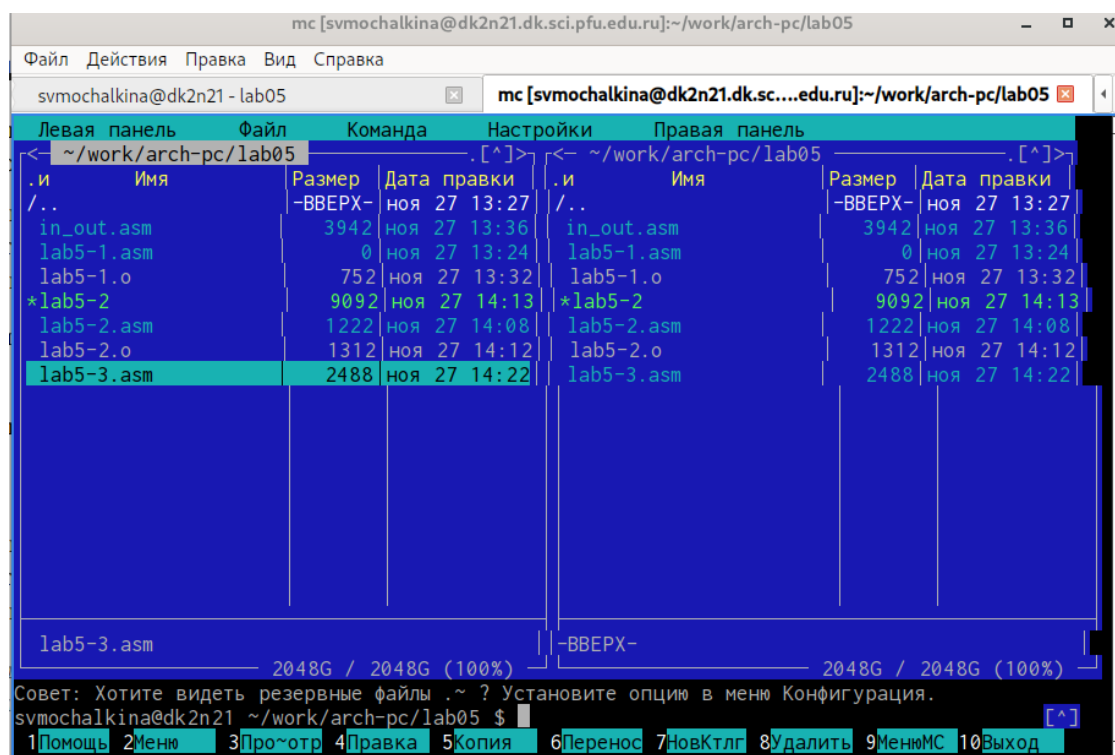
Исправленный текст программы (рис. 9)

```
svmochalkina@dk2n21 ~/work/arch-pc/lab05 $ nasm -f elf lab5-2.asm
svmochalkina@dk2n21 ~/work/arch-pc/lab05 $ ld -m elf_i386 -o lab5-2 lab5-2.o
svmochalkina@dk2n21 ~/work/arch-pc/lab05 $ ./lab5-2
Введите строку: Mochalkina Sofia Vasilievna
svmochalkina@dk2n21 ~/work/arch-pc/lab05 $
```

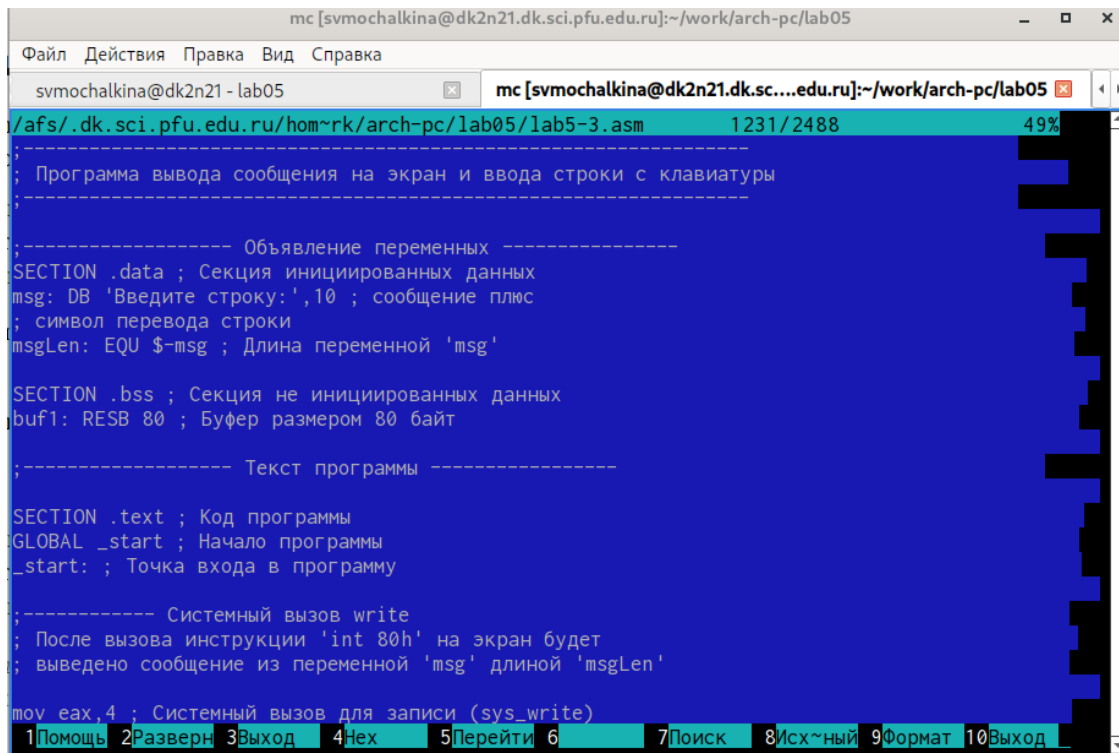
Работа файла lab5-2.asm(рис.10)

10. Самостоятельная работа.

11. Создаю копию файла lab5-1.asm.(рис 11) Вношу 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. (рис12)



Копия файла lab5-1.asm.(рис 11)



```
mc [svmochalkina@dk2n21.dk.sci.pfu.edu.ru]:~/work/arch-pc/lab05
Файл Действия Правка Вид Справка
svmochalkina@dk2n21 - lab05 mc [svmochalkina@dk2n21.dk.sc....edu.ru]:~/work/arch-pc/lab05
/afs/.dk.sci.pfu.edu.ru/home~rk/arch-pc/lab05/lab5-3.asm 1231/2488 49%
;-----
; 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 -----

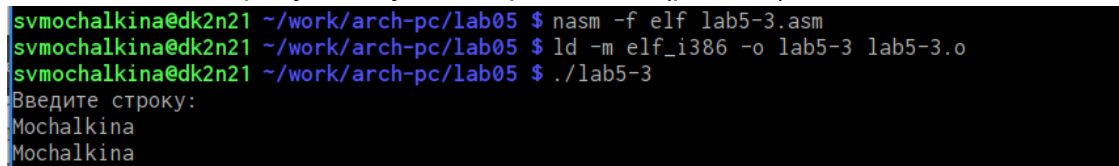
SECTION .text ; Код программы
GLOBAL _start ; Начало программы
_start: ; Точка входа в программу

;----- 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)
```

Изменения

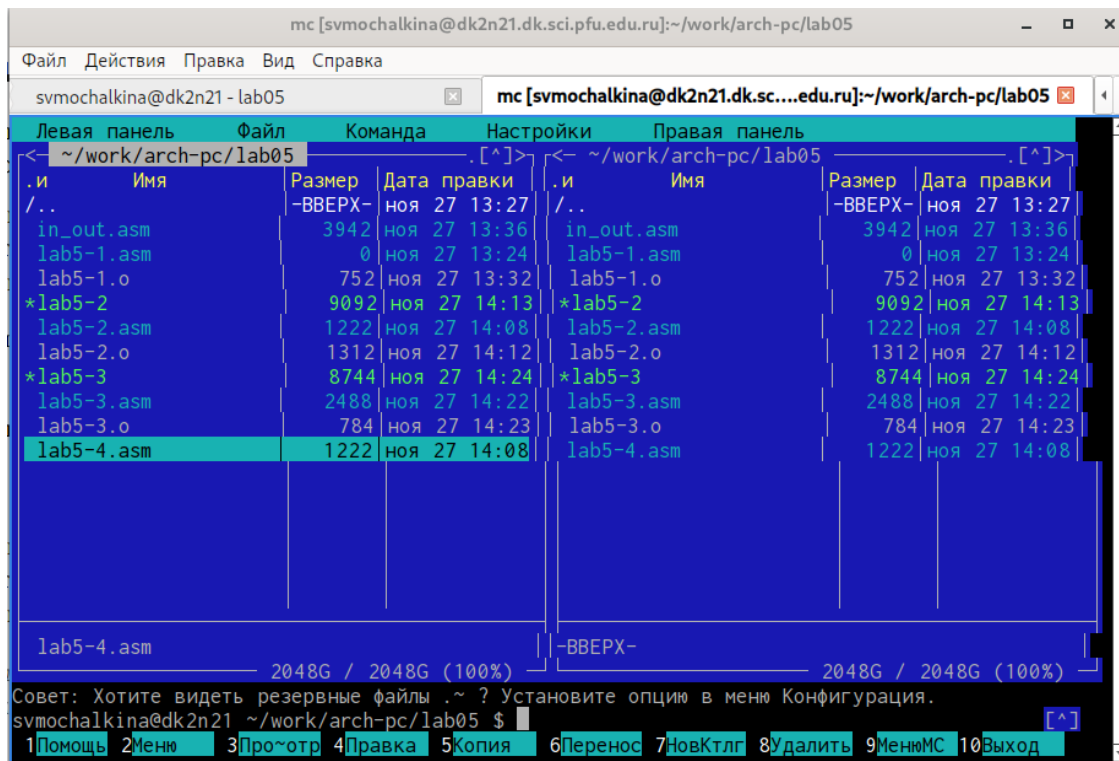
2. Получаю исполняемый файл и проверяю его работу. На приглашение ввести строку ввожу свою фамилию. (рис. 13)



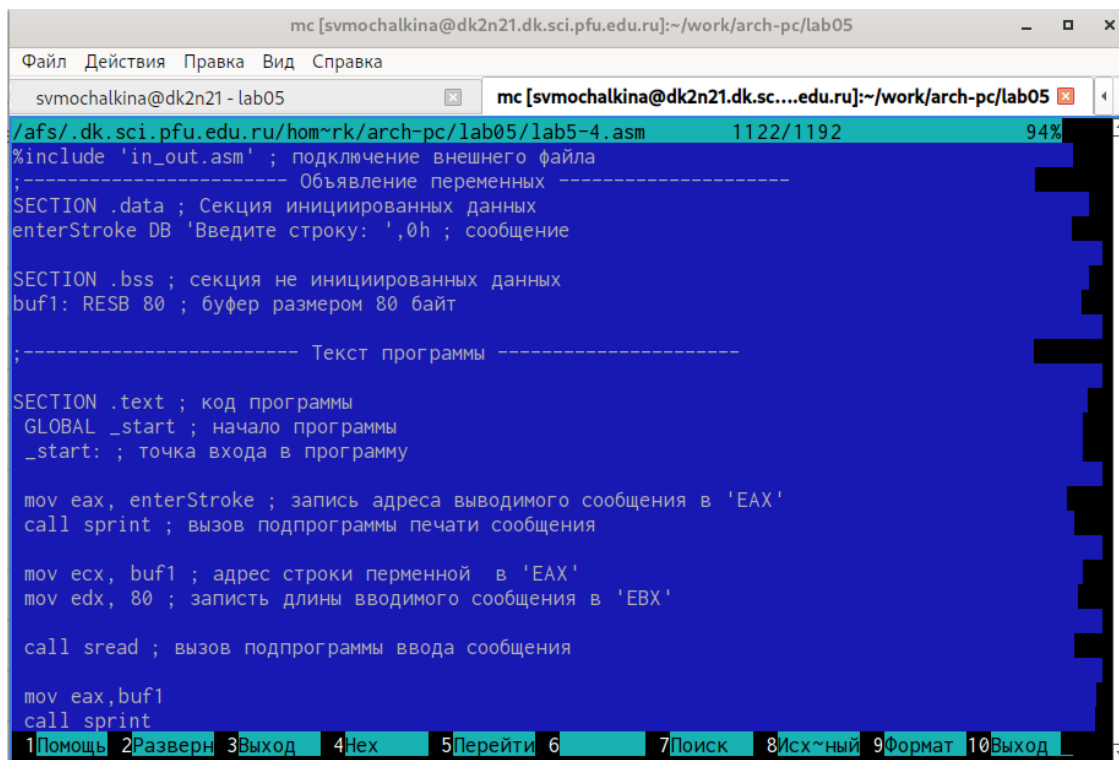
```
svmochalkina@dk2n21 ~/work/arch-pc/lab05 $ nasm -f elf lab5-3.asm
svmochalkina@dk2n21 ~/work/arch-pc/lab05 $ ld -m elf_i386 -o lab5-3 lab5-3.o
svmochalkina@dk2n21 ~/work/arch-pc/lab05 $ ./lab5-3
Введите строку:
Mochalkina
Mochalkina
```

Работа файла lab5-3 (рис. 13)

3. Создаю копию файла lab5-2.asm.(рис. 14) Исправляю 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. (рис. 15)



Копия файла lab5-2.asm. (рис 14)



Изменения

4. Создаю исполняемый файл и проверяю его работу.

```
svmochalkina@dk2n21 ~/work/arch-pc/lab05 $ nasm -f elf lab5-4.asm
svmochalkina@dk2n21 ~/work/arch-pc/lab05 $ ld -m elf_i386 -o lab5-4 lab5-4.o
svmochalkina@dk2n21 ~/work/arch-pc/lab05 $ ./lab5-4
Введите строку: Mochalkina
Mochalkina
svmochalkina@dk2n21 ~/work/arch-pc/lab05 $
```

Название рисунка

Выводы

В процессе выполнения лабораторной работы я ознакомилась со структурой программы на языке ассемблера NASM

Список литературы

1. GDB: The GNU Project Debugger. — URL: <https://www.gnu.org/software/gdb/>.
2. GNU Bash Manual. — 2016. — URL: <https://www.gnu.org/software/bash/manual/>.
3. Midnight Commander Development Center. — 2021. — URL: <https://midnight-commander.org/>.
4. NASM Assembly Language Tutorials. — 2021. — URL: <https://asmtutor.com/>.
5. Newham C. Learning the bash Shell: Unix Shell Programming. — O'Reilly Media, 2005. — 354 с. — (In a Nutshell). — ISBN 0596009658. — URL: <http://www.amazon.com/Learning-bash-Shell-Programming-Nutshell/dp/0596009658>.
6. Robbins A. Bash Pocket Reference. — O'Reilly Media, 2016. — 156 с. — ISBN 978-1491941591.
7. The NASM documentation. — 2021. — URL: <https://www.nasm.us/docs.php>.
8. Zarrelli G. Mastering Bash. — Packt Publishing, 2017. — 502 с. — ISBN 9781784396879.
9. Колдаев В. Д., Лупин С. А. Архитектура ЭВМ. — М. : Форум, 2018.
10. Куляс О. Л., Никитин К. А. Курс программирования на ASSEMBLER. — М. : Солон-Пресс,
11. Новожилов О. П. Архитектура ЭВМ и систем. — М. : Юрайт, 2016.
12. Расширенный ассемблер: NASM. — 2021. — URL: <https://www.opennet.ru/docs/RUS/nasm/>.
13. Робачевский А., Немнюгин С., Стесик О. Операционная система UNIX. — 2-е изд. — БХВ- Петербург, 2010. — 656 с. — ISBN 978-5-94157-538-1.
14. Столяров А. Программирование на языке ассемблера NASM для ОС Unix. — 2-е изд. — М. : МАКС Пресс, 2011. — URL: http://www.stolyarov.info/books/asm_unix.
15. Таненбаум Э. Архитектура компьютера. — 6-е изд. — СПб. : Питер, 2013. — 874 с. — (Классика Computer Science).

16. Таненбаум Э., Бос Х. Современные операционные системы. — 4-е изд. — СПб. : Питер,
17. — 1120 с. — (Классика Computer Science)