

Отчёт по лабораторной работе №5

Дисциплина: архитектура компьютера

Аветисян Алина Эдуардовна НММбд-01-23

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
3	Выполнение заданий для самостоятельной работы	14
3.0.1	Задание №1	14
3.0.2	Задание №2	16
4	Выводы	18

Список иллюстраций

2.1	Ввела mc	6
2.2	Перехожу в каталог	6
2.3	Создаю каталог	7
2.4	Перехожу в каталог	7
2.5	Создаю lab5-1.asm	8
2.6	Открыла файл для редактирования и ввела текст	8
2.7	9
2.8	Открыла файл для просмотра	9
2.9	Оттранслировала текст программы и выполнила компоновку объектного файла	10
2.10	Скачала файл in_out.asm	10
2.11	Копирование файла	11
2.12	11
2.13	Создание копии	12
2.14	Изменение содержимого файла	12
2.15	Исполнение файла	13
2.16	Замена программы	13
2.17	Создание исполняемого файла и проверка	13
3.1	Создание копии файла	14
3.2	15
3.3	Редактирование файла	15
3.4	Исполнение файла	16
3.5	Создание копии файла	16
3.6	Редактирование файла	16
3.7	Исполнение файла	17

Список таблиц

1 Цель работы

Целью данной лабораторной работы является приобретение практических навыков работы в Midnight Commander и освоение инструкций языка ассемблера `mov` и `int`.

2 Выполнение лабораторной работы

Открываю Midnight Commander, введя `mc`

```
aeavetisyan@dk8n64 ~ $ mc
```

Рис. 2.1: Ввела `mc`

Перехожу в каталог `~/work/arch-rc`

Левая панель				Файл	Команда	Настройки	Правая панель			
< ~/work/arch-pc							> .[^>]			
.и		Имя		Размер	Время правки		.и		Имя	
/..				-ВВЕРХ-	окт 18 12:06		/..		-ВВЕРХ- сен 20 18:01	
/lab04				2048	окт 18 13:17		/.cache		2048 сен 13 13:24	
							/.config		4096 ноя 8 11:58	
							/.gnupg		2048 ноя 8 11:58	
							/.local		2048 сен 6 12:06	
							/.mozilla		2048 сен 13 12:11	
							/.ssh		2048 окт 11 12:36	
							/.texlive2022		2048 окт 11 12:58	
							/.texlive2023		2048 окт 12 09:56	
							/dir1		2048 сен 12 20:33	
							/dir2		2048 сен 12 20:33	
							/dir3		2048 сен 12 20:33	
							/lab		2048 сен 15 14:46	

Рис. 2.2: Перехожу в каталог

Создаю каталог `lab05` с помощью клавиши `F7`

Создать новый каталог

Введите имя каталога:

lab05

[< Дальше >] [Прервать]

Рис. 2.3: Создаю каталог

[Перехожу в каталог](#)

```
< ~ /work/arch-pc/lab05 .[>
.и Имя Размер Время правки .и Имя Размер Время правки
/.. -BBERX- ноя 8 12:03 /.. -BBERX- сен 20 18:01
/cache 2048 сен 13 13:24
/config 4096 ноя 8 12:03
/gnupg 2048 ноя 8 11:58
/local 2048 сен 6 12:06
/mozilla 2048 сен 13 12:11
/ssh 2048 окт 11 12:36
/texlive2022 2048 окт 11 12:58
/texlive2023 2048 окт 12 09:56
/dir1 2048 сен 12 20:33
/dir2 2048 сен 12 20:33
/dir3 2048 сен 12 20:33
/lab 2048 сен 15 14:46
/parentdir 2048 сен 13 13:07
/parentdir1 2048 сен 15 13:41
/parentdir2 2048 сен 13 13:13
/parentdir3 2048 сен 13 13:17
/public 2048 сен 6 12:58
~public_html 18 сен 11 20:09
/test 2048 сен 6 13:06
/tmp 2048 сен 15 13:45

-BBERX- -BBERX-
2048G/2048G (100%) 2048G/2048G (100%)

Совет: Хотите простую оболочку? Нажмите C-o, и снова C-o для возврата в MC.
aeavetisyan@dk8n64 ~ /work/arch-pc/lab05 $ touch lab5-1.asm
```

Рис. 2.4: Перехожу в каталог

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

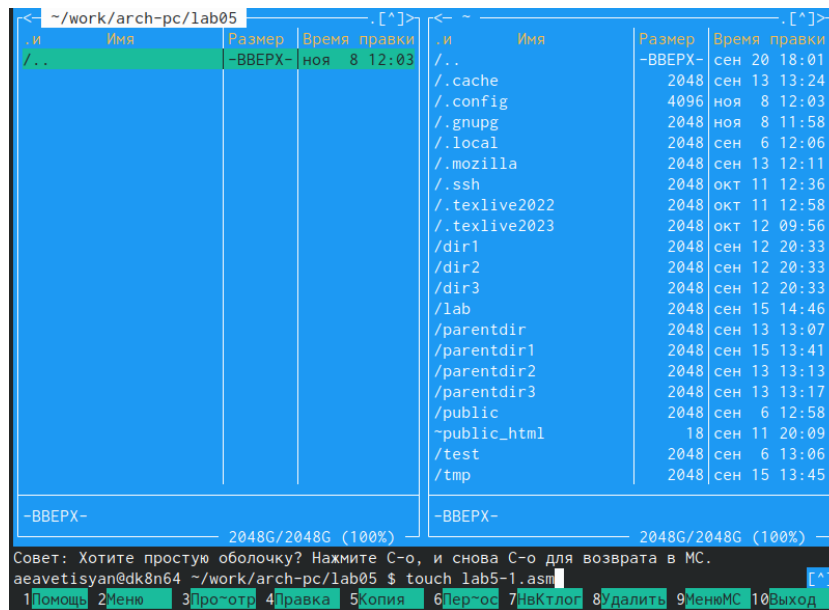
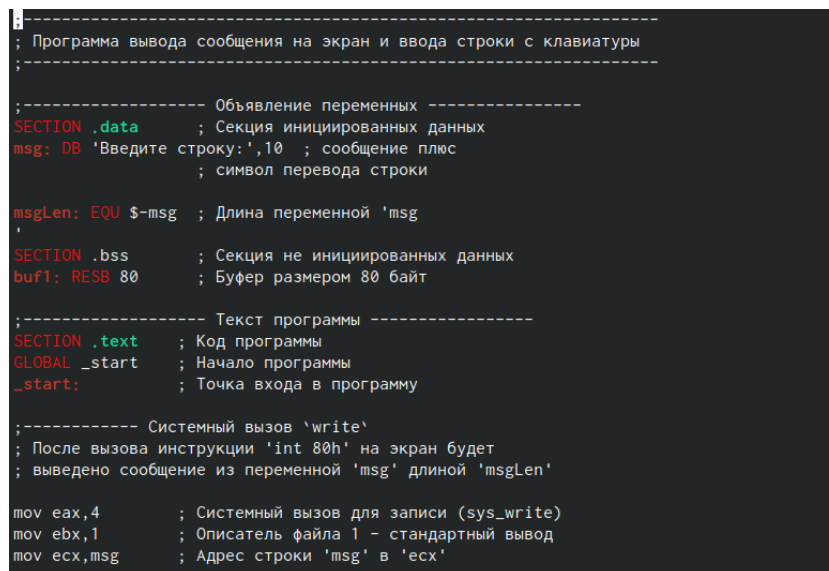


Рис. 2.5: Создаю lab5-1.asm

С помощью функциональной клавиши F4 открываю файл lab5-1.asm для редактирования во встроенном редакторе nano. Ввожу текст программы из листинга




```

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       ;Descriptor 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' -----
; После вызова инструкции 'int 80h' программа завершит работу

mov eax,1        ; Системный вызов для выхода (sys_exit)
mov ebx,0        ; Выход с кодом возврата 0 (без ошибок)
int 80h          ; Вызов ядра

```

Рис. 2.7:

С помощью функциональной клавиши F3 открыла файл lab5-1.asm для просмотра. Убеждаюсь, что файл содержит текст программы

```

/afs/.dk.sci.pfu.edu.ru/home/k/arch-pc/lab05/lab5-1.asm 1535/2571 59%
;-----
; 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

```

Рис. 2.8: Открыла файл для просмотра

Оттранслирую текст программы lab5-1.asm в объектный файл. Выполняю компоновку объектного файла и запускаю получившийся исполняемый файл

```

aeavetisyan@k8n64 ~/work/arch-pc/lab05 $ nasm -f elf lab5-1.asm
aeavetisyan@k8n64 ~/work/arch-pc/lab05 $ ld -m elf_i386 -o lab5-1 lab5-1.o
aeavetisyan@k8n64 ~/work/arch-pc/lab05 $ ./lab5-1
Введите строку:
Аветисян Алина Эдуардовна

```

Рис. 2.9: Оттранслировала текст программы и выполнила компоновку объектного файла

Скачиваю файл in_out.asm со страницы курса в ТУИС (сохранился в каталоге “Загрузки”)

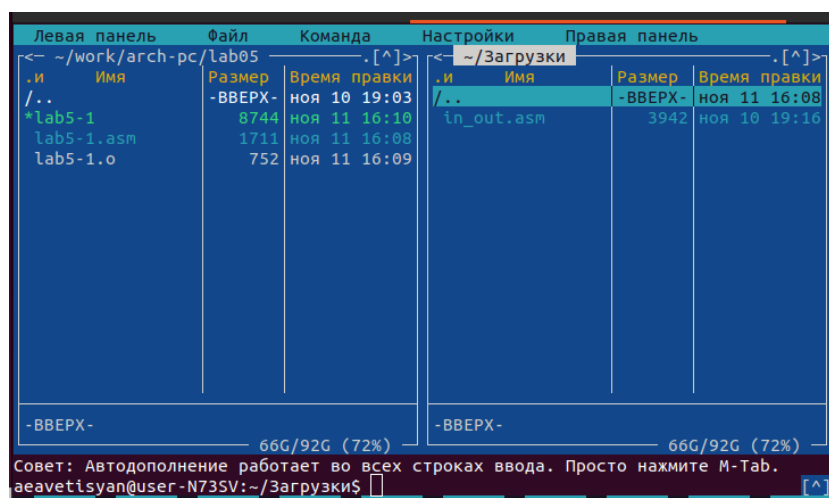


Рис. 2.10: Скачала файл in_out.asm

Копирую файл in_out.asm из каталога “Загрузки” в созданный каталог lab05 с помощью клавиши F5

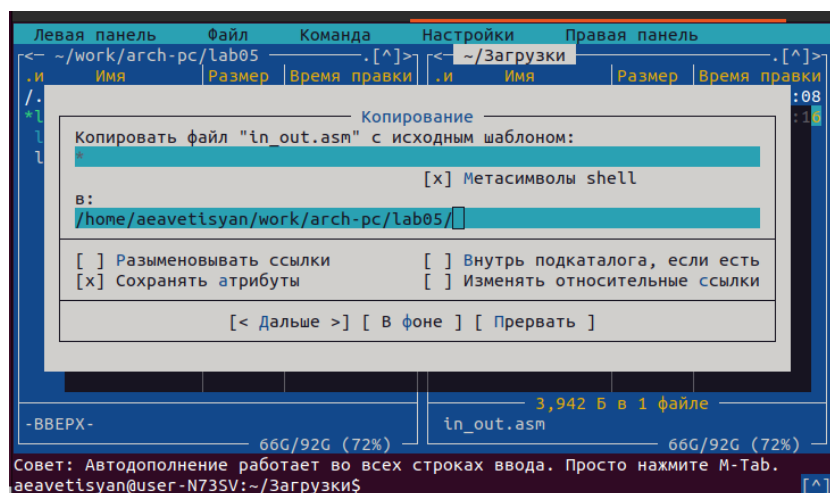


Рис. 2.11: Копирование файла

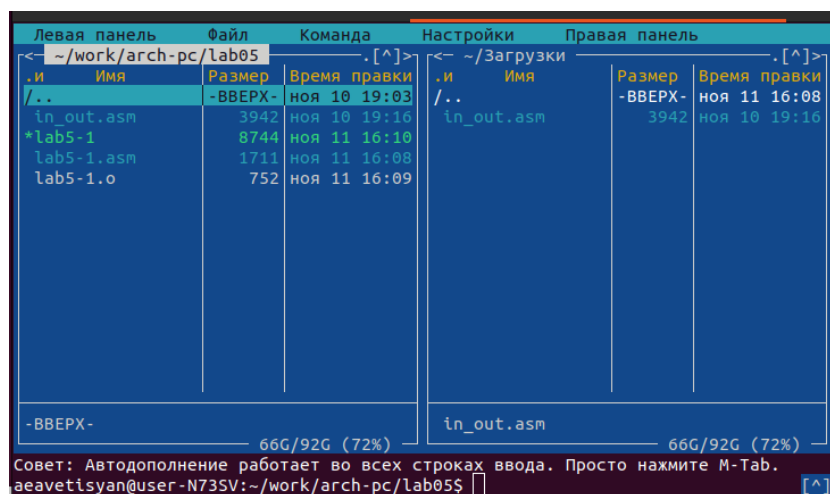


Рис. 2.12:

Создаю копию файла lab5-1.asm с именем lab5-2.asm

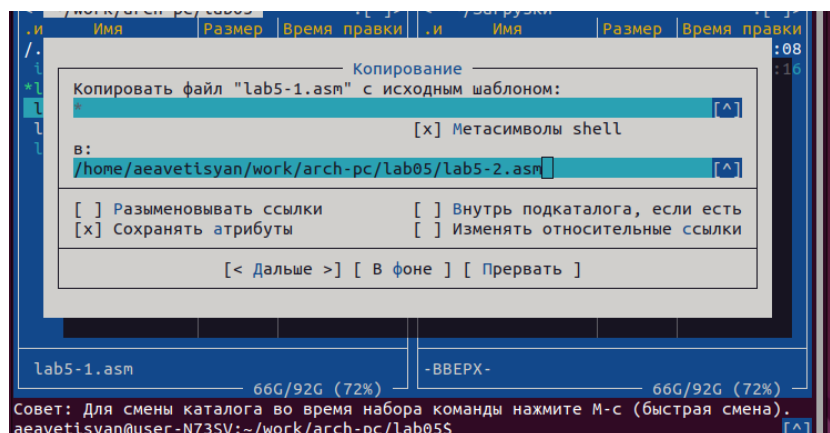


Рис. 2.13: Создание копии

Изменяю содержимое файла lab5-2.asm во встроенном редакторе nano, чтобы были использованы подпрограммы из внешнего файла in_out.asm

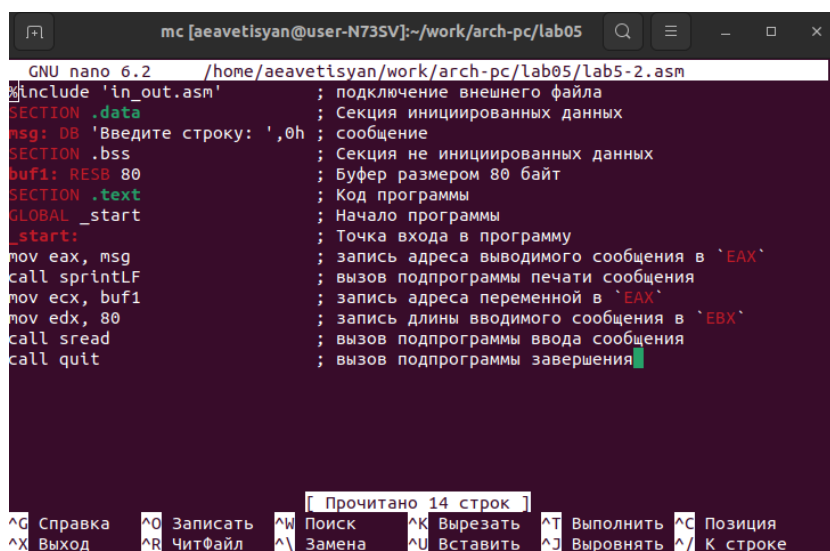


Рис. 2.14: Изменение содержимого файла

Транслирую текст программы файла в объектный файл; создался объектный файл; выполняю компоновку объектного файла; создался исполняемый файл; запускаю исполняемый файл

```

aeavetisyan@user-N73SV:~/work/arch-pc/lab05$ nasm -f elf lab5-2.asm
aeavetisyan@user-N73SV:~/work/arch-pc/lab05$ ld -m elf_i386 -o lab5-2 lab5-2.o
aeavetisyan@user-N73SV:~/work/arch-pc/lab05$ ./lab5-2
Введите строку:
Аветисян Алина Эдуардовна

```

Рис. 2.15: Исполнение файла

В файле lab5-2.asm заменяю подпрограмму sprintLF на sprint

```

GNU nano 6.2 /home/aeavetisyan/work/arch-pc/lab05/lab5-2.asm *
#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 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

```

Рис. 2.16: Замена программы

Создаю исполняемый файл и проверяю его работу(используя подпрограмму sprintLF запрашивается ввод с новой строки,а при sprint запрашивается ввод без переноса на новую строку)(рис. [??]).

```

aeavetisyan@user-N73SV:~/work/arch-pc/lab05$ nasm -f elf lab5-2.asm
aeavetisyan@user-N73SV:~/work/arch-pc/lab05$ ld -m elf_i386 -o lab5-2 lab5-2.o
aeavetisyan@user-N73SV:~/work/arch-pc/lab05$ ./lab5-2
Введите строку: Аветисян Алина Эдуардовна
aeavetisyan@user-N73SV:~/work/arch-pc/lab05$

```

Рис. 2.17: Создание исполняемого файла и проверка

3 Выполнение заданий для самостоятельной работы

3.0.1 Задание №1

Копирую файл lab5-1.asm с именем lab5-1-1.asm

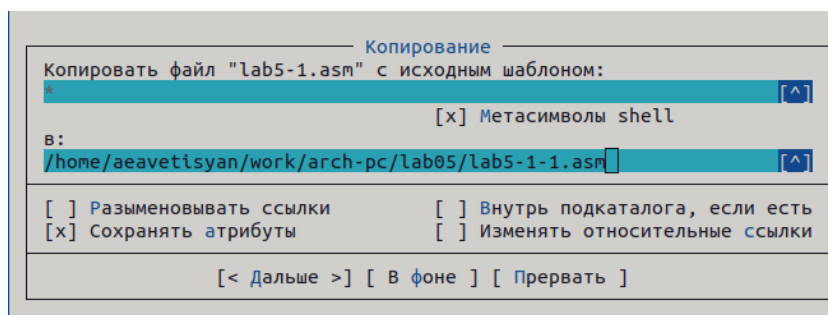


Рис. 3.1: Создание копии файла

/..	-BBE
in_out.asm	3
*lab5-1	8
*lab5-1-1	8
lab5-1-1.asm	1
lab5-1-1.o	

Рис. 3.2:

С помощью функциональной клавиши F4 открываю созданный файл для редактирования. Изменяю программу так, чтобы кроме вывода приглашения и запроса ввода, она выводила вводимую пользователем строку

```
GNU nano 6.2 /home/aeavetisyan/work/arch-pc/lab05/lab5-1-1.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,1
mov ebx,0
int 80h
```

Рис. 3.3: Редактирование файла

Создаю объектный файл lab6-1-1.o, отдаю его на обработку компоновщику, получаю исполняемый файл lab6-1-1, запускаю полученный исполняемый файл. Программа запрашивает ввод, ввожу свои ФИО, далее программа выводит введенные мною данные

```
aeavetisyan@user-N735V:~/work/arch-pc/lab05$ nasm -f elf lab5-1-1.asm
aeavetisyan@user-N735V:~/work/arch-pc/lab05$ ld -m elf_i386 -o lab5-1-1 lab5-1-1.o
aeavetisyan@user-N735V:~/work/arch-pc/lab05$ ./lab5-1-1
Введите строку:
Аветисян Алина Эдуардовна
Аветисян Алина Эдуардовна
aeavetisyan@user-N735V:~/work/arch-pc/lab05$
```

Рис. 3.4: Исполнение файла

3.0.2 Задание №2

Создаю копию файла lab5-2.asm с именем lab5-2-1.asm с помощью функциональной клавиши F5

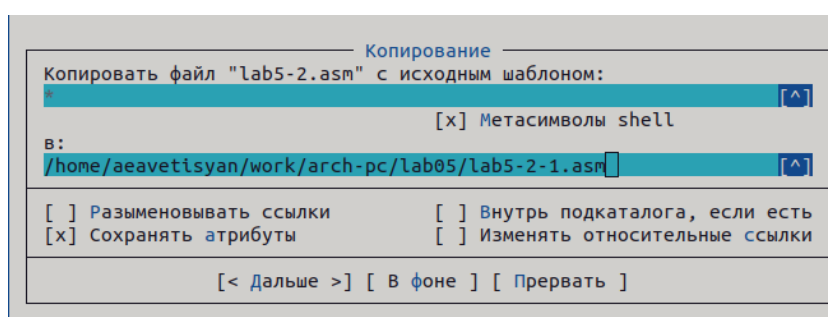


Рис. 3.5: Создание копии файла

С помощью функциональной клавиши F4 открываю созданный файл для редактирования. Изменяю программу так, чтобы кроме вывода приглашения и запроса ввода, она выводила вводимую пользователем строку

```
GNU nano 6.2 /home/aeavetisyan/work/arch-pc/lab05/lab5-2-1.asm
#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
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
;-----Вывод на экран-----
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
```

Рис. 3.6: Редактирование файла

Создаю объектный файл lab5-2-1.o, отдаю его на обработку компоновщику, получаю исполняемый файл lab5-2-1, запускаю полученный исполняемый файл.

Программа запрашивает ввод без переноса на новую строку, ввожу свои ФИО, далее программа выводит введенные мною данные

```
aeavetisyan@user-N73SV:~/work/arch-pc/lab0$ nasm -f elf lab5-2-1.asm
aeavetisyan@user-N73SV:~/work/arch-pc/lab0$ ld -m elf_i386 -o lab5-2-1 lab5-2-1.o
aeavetisyan@user-N73SV:~/work/arch-pc/lab0$ ./lab5-2-1
Введите строку: Аветисян Алина Эдуардовна
Аветисян Алина Эдуардовна
aeavetisyan@user-N73SV:~/work/arch-pc/lab0$
```

Рис. 3.7: Исполнение файла

4 Выводы

Я приобрела практические навыки работы в Midnight Commander и освоила инструкции `mov` и `int` в языке ассемблер.