

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра прикладной информатики и теории вероятностей

ОТЧЕТ

ПО ЛАБОРАТОРНОЙ РАБОТЕ № 5

дисциплина: *Архитектура компьютера*

Студент: *Сухова Арина Евгеньевна*

Группа: НПИбд-02-25

МОСКВА

2025 г.

Содержание

1. Цель работы.....	2
2. Задание.....	2
3. Выполнение лабораторной работы.....	2
3.1 Порядок выполнения лабораторной работы.....	2
3.2 Задание для самостоятельной работы.....	11
4. Выводы.....	14

1. Цель работы

Освоить инструкции языка ассемблера mov. Приобрести знания использования Midnight Commander.

2. Задание

Написать 2 программы по примеру и впоследствии изменить их по условию.

3. Выполнение лабораторной работы

3.1 Порядок выполнения лабораторной работы

Открываем Midnight Commander (рис.1)

Правая панель			
< ~ .[^]>			
.и	Имя	Размер	Дата правки
/..		-ВВЕРХ-	сен 16 01:13
/.cache		542	окт 27 15:37
/.config		384	окт 27 15:37
/.local		20	сен 16 01:13
/.mozilla		48	сен 22 18:07
/.ssh		100	сен 28 23:14
/.var		6	сен 22 18:21
/parentdir		0	сен 24 21:22
/tmp		0	сен 24 21:43
/work		24	окт 26 17:02
/Видео		0	сен 16 01:13
/Документы		0	сен 16 01:13
/Загрузки		248	окт 20 20:33
/Изображения		50	сен 22 18:10
/Музыка		0	сен 16 01:13
/Общедоступные		0	сен 16 01:13
/Рабочий стол		0	сен 16 01:13
/Шаблоны		0	сен 16 01:13
.bash_history		3028	окт 27 16:15
.bash_logout		18	ноя 8 2024
.bash_profile		144	ноя 8 2024
.bashrc		558	окт 20 20:41
.gitconfig		121	сен 28 22:56
.vboxclient-cl~y2-control.pid		5	окт 26 16:36
.vboxclient-cl~y2-service.pid		5	окт 26 16:36
.vboxclient-dr~y2-control.pid		5	окт 26 16:36
.vboxclient-ho~y2-control.pid		5	окт 26 16:36
.vboxclient-se~y2-control.pid		5	окт 26 16:36
quarto-1.4.554~x-amd64.tar.gz		115750K	мая 2 2024

Рис 1. Вводим в консоль команду `tc`

Переходим в каталог, созданный при выполнении 4ЛБ (рис. 2)

Правая панель			
< ~/work/arch-pc .[^]>			
.и	Имя	Размер	Дата правки
/..		-ВВЕРХ-	окт 26 17:02
/lab04		104	окт 26 17:49

Рис. 2 Переходим в каталог

Создаем каталог `lab05` функциональной клавишей `F7`, создаем файл `lab5-1.asm` (рис. 3)

Правая панель			
< ~ /work/arch-pc/lab05 .[^]>			
.и	Имя	Размер	Дата правки
/..		-ВВЕРХ-	окт 27 15:43
	lab5-1.asm	2506	окт 27 16:43

Рис. 3 Воспользуемся командой touch

Открываем файл для редактирования и заполняем его по листингу (рис. 4)

```

/home/aesukhova/work/arch-pc/lab05/lab5-1.asm 2506/2506 100
;-----
; 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                 ;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. 4 Открываем файл функциональной клавишей, заполняем и сохраняем

Открываем файл для просмотра и убеждаемся, что файл содержит текст программы (рис.5)

```

/home/aesukhova/work/arch-pc/lab05/lab5-1.asm 2506/2506 100
;-----
; 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 ;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. 5 Текст программы

Транслируем текст программы и запускаем исполняемый файл (рис.6)

```

aesukhova@fedora:~/work/arch-pc/lab05$ nasm -f elf lab5-1.asm
aesukhova@fedora:~/work/arch-pc/lab05$ ld -m elf_i386 -o lab5-1 lab5-1.o
aesukhova@fedora:~/work/arch-pc/lab05$ ./lab5-1
Введите строку:
Сухова Арина Евгеньевна

```

Рис. 6 Проверяем, как работает данная программа

Скачиваем файл со страницы ТУИС (рис.7)

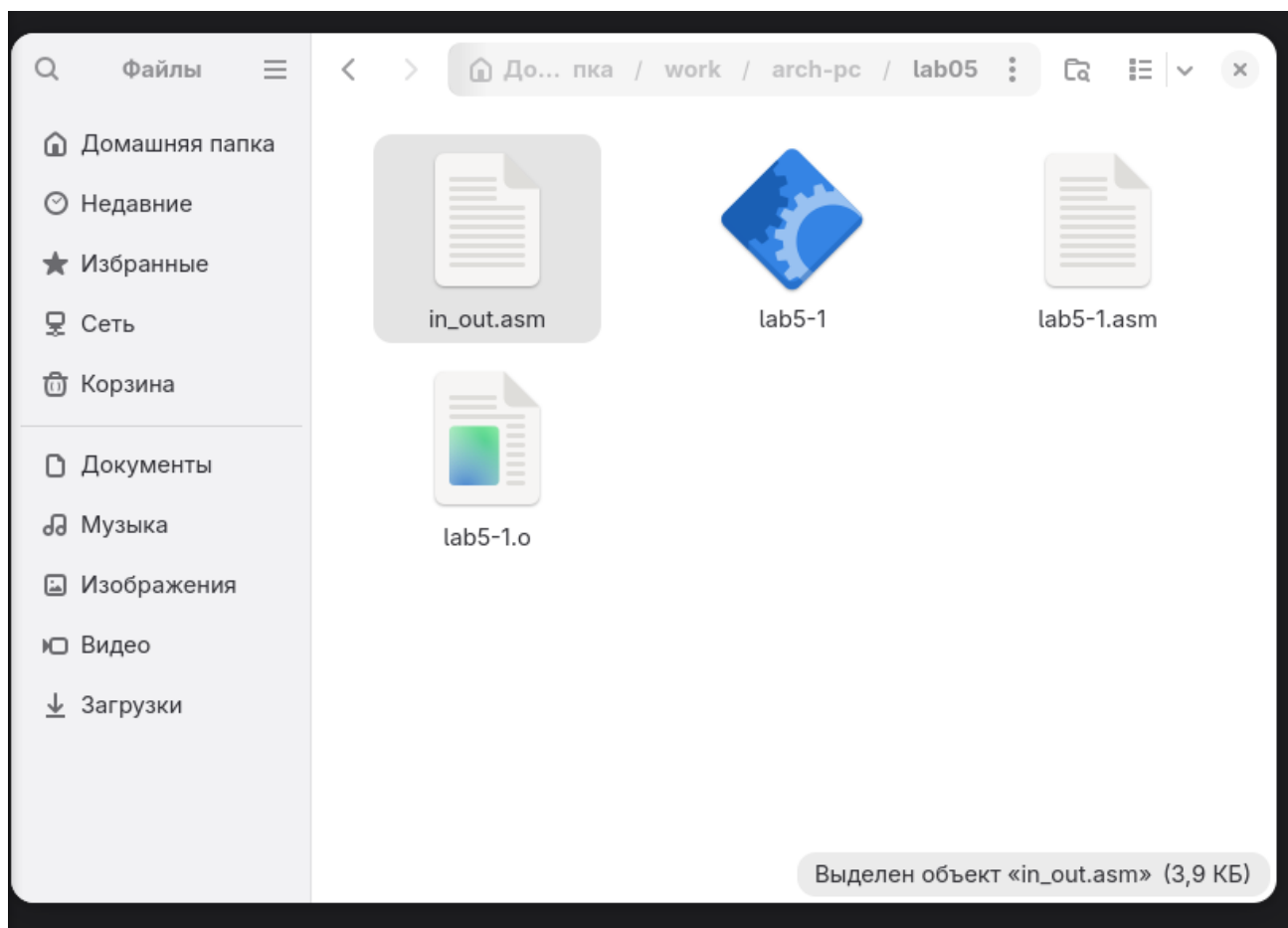


Рис. 7 Скачиваем файл

Копируем файл в нужную директорию (рис.8)

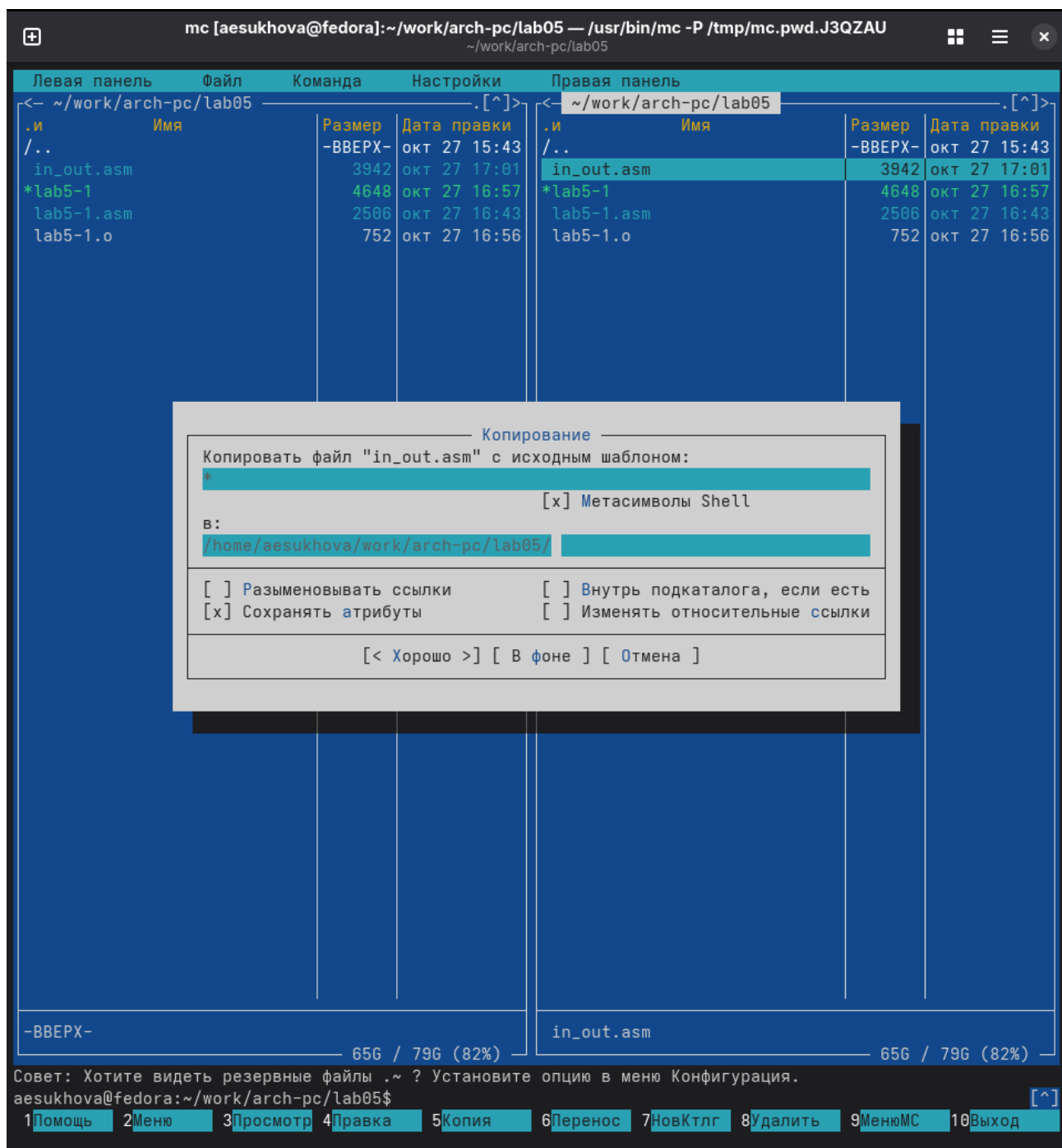


Рис. 8 Копируем скачанный файл

Создаем копию файла lab5-1.asm (рис. 9)

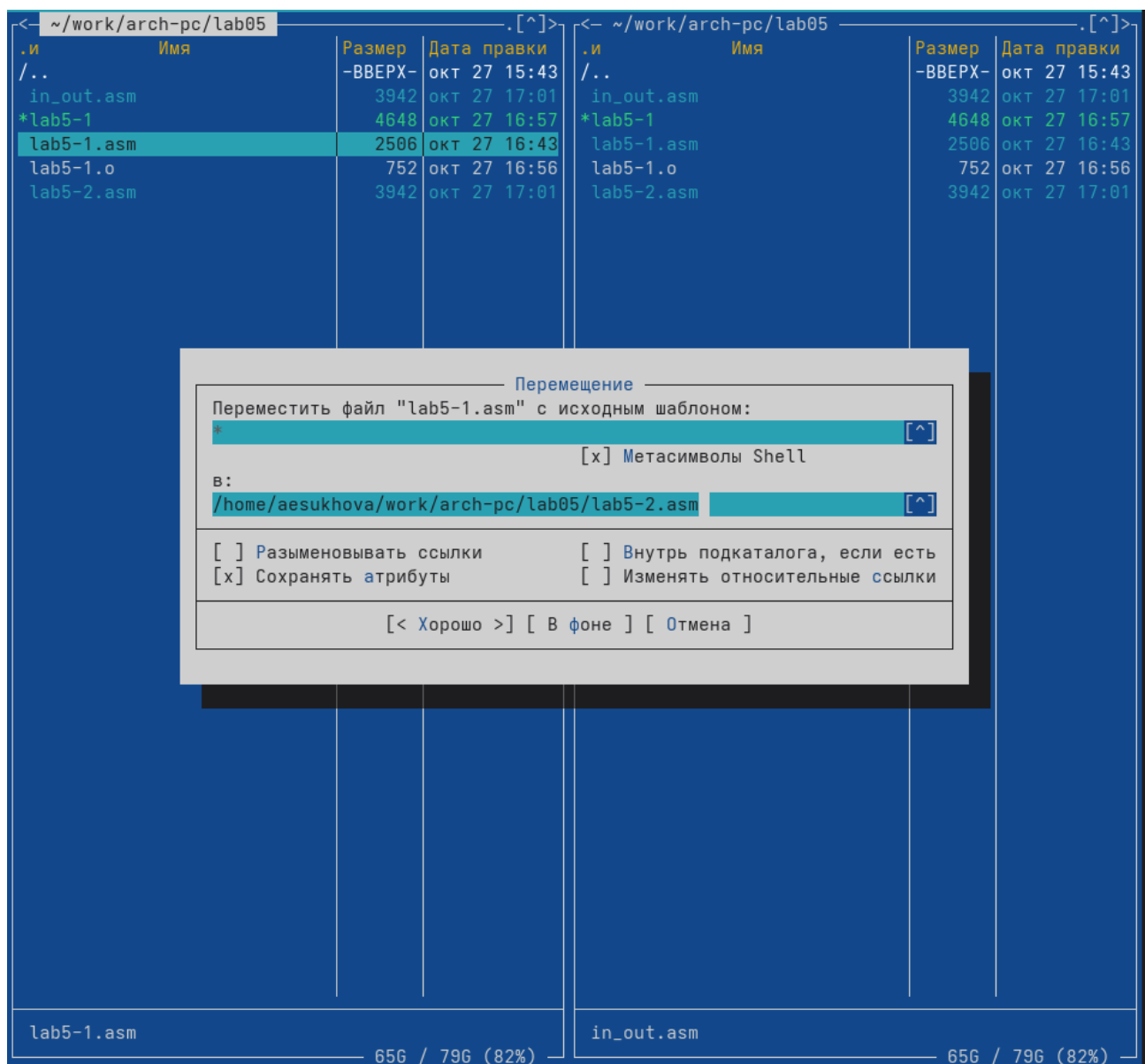


Рис. 9 Создаем копию файла клавишей F6

Проверяем созданный файл (рис. 10)

Левая панель		Файл		Команда		Настройки		Правая панель	
← ~/work/arch-pc/lab05						. [^]		← ~/work/arch-pc/lab05	
		Имя		Размер		Дата правки			
				-ВВЕРХ-		окт 27 15:43			
.и				4648		окт 27 16:57		.и	
/..				2506		окт 27 16:43		/..	
*lab5-1				752		окт 27 16:56		*lab5-1	
lab5-1.asm				3942		окт 27 17:01		lab5-1.asm	
lab5-1.o								lab5-1.o	
lab5-2.asm								lab5-2.asm	

```

aesukhova@fedora:~/work/arch-pc/lab05$ nasm -f elf lab5-2.asm
aesukhova@fedora:~/work/arch-pc/lab05$ ld -m elf_i386 -o lab5-2 lab5-2.o
aesukhova@fedora:~/work/arch-pc/lab05$ ./lab5-2
Введите строку:
Сухова Арина Евгеньевна
aesukhova@fedora:~/work/arch-pc/lab05$

```

Рис. 12 Смотрим, как сработала программа

Снова открываем файл для редактирования и меняем sprintf на printf (рис. 13)

```

lab5-2.asm      [-M--] 15 L:[ 1+16 17/ 22] *(883 /1287b) 0009 0x009  [*][X]
-----
; 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
SECTION .text<-><-----><----->; Код программы
GLOBAL _start<-><-----><----->; Начало программы
_start:<-><-----><----->; Точка входа в программу
....
mov eax, msg<-><-----><----->; запись адреса выводимого сообщения в `EAX`
call printf<-><-----><----->; 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. 13 Редактируем файл

Транслируем и запускаем файл (рис. 14)

```

aesukhova@fedora:~/work/arch-pc/lab05$ nasm -f elf lab5-2.asm
aesukhova@fedora:~/work/arch-pc/lab05$ ld -m elf_i386 -o lab5-2 lab5-2.o
aesukhova@fedora:~/work/arch-pc/lab05$ ./lab5-2
Введите строку: Сухова Арина Евгеньевна
aesukhova@fedora:~/work/arch-pc/lab05$

```

Рис. 14 Сравниваем с прошлой

3.2 Задание для самостоятельной работы

Левая панель		Файл	Команда	Настройки	Правая панель	
< ~ /work/arch-pc/lab05					< ~ /work/arch-pc/lab05	
.и	Имя	Размер	Дата правки		.и	Имя
/..		-ВВЕРХ-	окт 27 15:43		/..	
archc		2506	окт 27 16:43		archc	2506 окт 27 16:43
in_out.asm		3942	окт 27 17:01		in_out.asm	3942 окт 27 17:01
*lab5-1		4648	окт 27 16:57		*lab5-1	4648 окт 27 16:57
lab5-1.1.asm		2507	окт 28 00:08		lab5-1.1.asm	2507 окт 28 00:08
lab5-1.asm		2506	окт 27 16:43		lab5-1.asm	2506 окт 27 16:43
lab5-1.o		752	окт 27 16:56		lab5-1.o	752 окт 27 16:56
*lab5-2		4996	окт 27 23:59		*lab5-2	4996 окт 27 23:59
lab5-2.asm		1287	окт 27 23:57		lab5-2.asm	1287 окт 27 23:57
lab5-2.o		1312	окт 27 23:57		lab5-2.o	1312 окт 27 23:57
*lab5-3		4648	окт 28 00:56		*lab5-3	4648 окт 28 00:56
lab5-3.asm		2583	окт 28 00:55		lab5-3.asm	2583 окт 28 00:55
lab5-3.o		784	окт 28 00:55		lab5-3.o	784 окт 28 00:55

Редактируем файл, чтобы введенный текст с клавиатуры выводился в консоль (рис. 16)

```

/home/aesukhova/work/arch-pc/lab05/lab5-3.asm 2583/2583 100%
;-----
; 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

    mov eax,4
    mov ebx,1
    mov ecx,buf1
    mov edx,80
    int 80h
;----- 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. 16 Редактируем файл

Транслируем файл и запускаем программу (рис. 17)

```

aesukhova@fedora:~/work/arch-pc/lab05$ nasm -f elf lab5-3.asm
aesukhova@fedora:~/work/arch-pc/lab05$ ld -m elf_i386 -o lab5-3 lab5-3.o
aesukhova@fedora:~/work/arch-pc/lab05$ ./lab5-3
Введите строку:
Сухова Арина Евгеньевна
Сухова Арина Евгеньевна

```

Создаем копию файла lab5-2.asm и называем его lab5-4.asm (рис.18)

Создаем копию файла lab5-2.asm и называем его lab5-4.asm (рис.18)

Рис. 18 Создаем копию файла lab5-2.asm

Редактируем файл, чтобы введенный текст выводился в консоль (рис. 19)

```
/home/aesukhova/work/arch-pc/lab05/lab5-4.asm 1261/1261 100%
;-----
; 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
    mov eax,buf1
    call sprintf
    call quit
```

Рис. 19 Редактируем файл

Транслируем файл и запускаем программу (рис. 20)

```
aesukhova@fedora:~/work/arch-pc/lab05$ nasm -f elf lab5-4.asm
aesukhova@fedora:~/work/arch-pc/lab05$ ld -m elf_i386 -o lab5-4 lab5-4.o
aesukhova@fedora:~/work/arch-pc/lab05$ ./lab5-4
Введите строку:
Сухова Арина Евгеньевна
Сухова Арина Евгеньевна
```

Рис. 20 Проверяем правильность написания программы

4. Выводы

Мы приобрели навыки работы с Midnight Commander и освоили инструкцию mov.