

Отчёт по лабораторной работе №5

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

Данил Евгеньевич Овчиников

Содержание

1 Цель работы

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

2 Выполнение лабораторной работы

1. Откроем Midnight Commander.

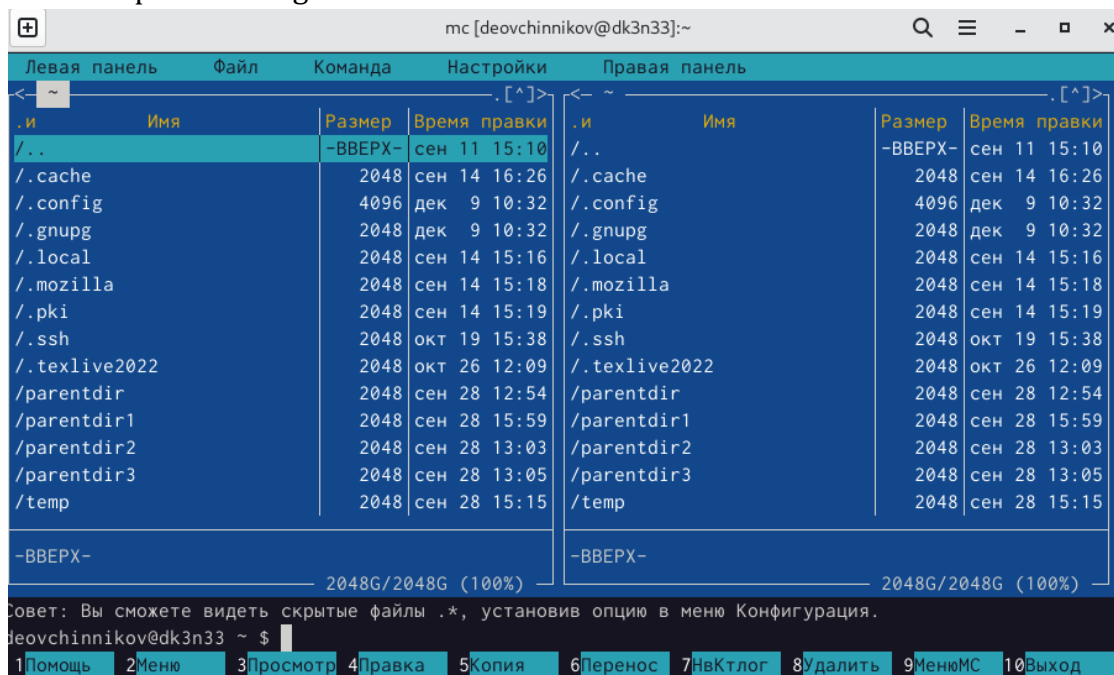


Рис. 1: Открытие Midnight Commander

2. Перейдём в каталог ~/work/arch-рс созданный при выполнении лабораторной работы No4 и создадим папку lab05, затем перейдём в созданный каталог.

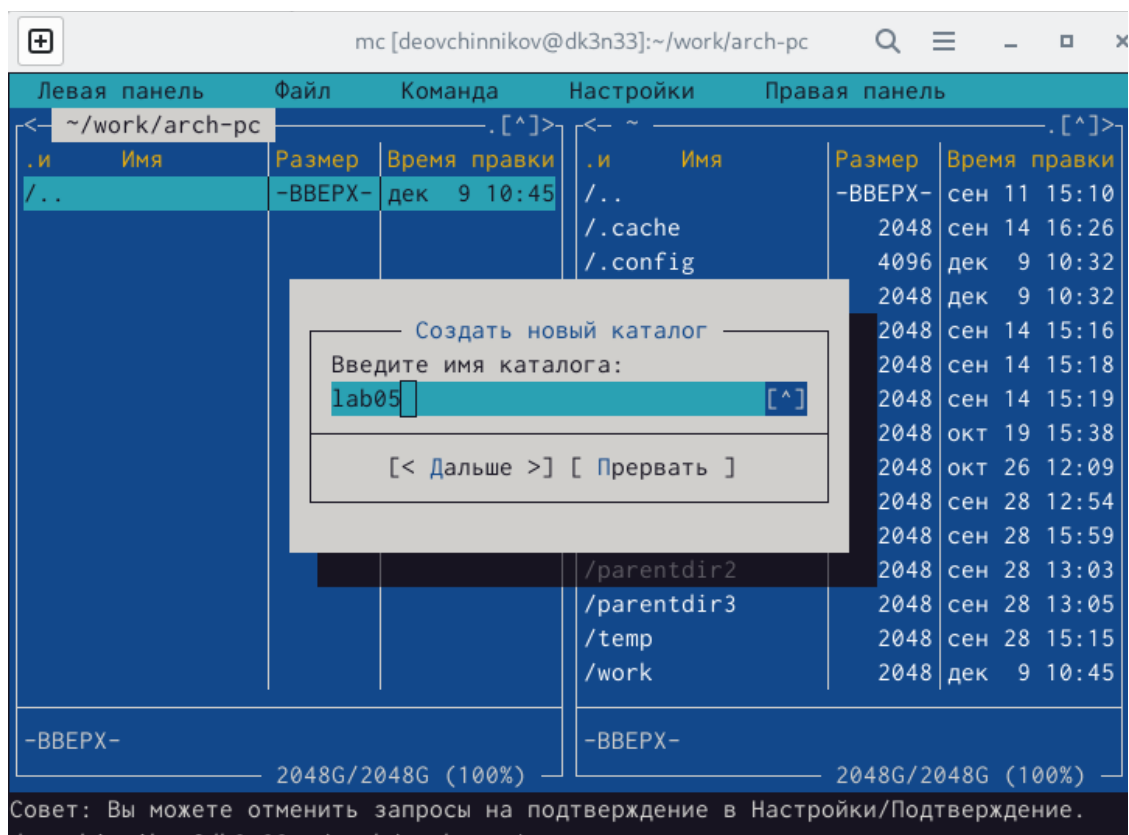


Рис. 2: Переход в каталог и создание папки

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

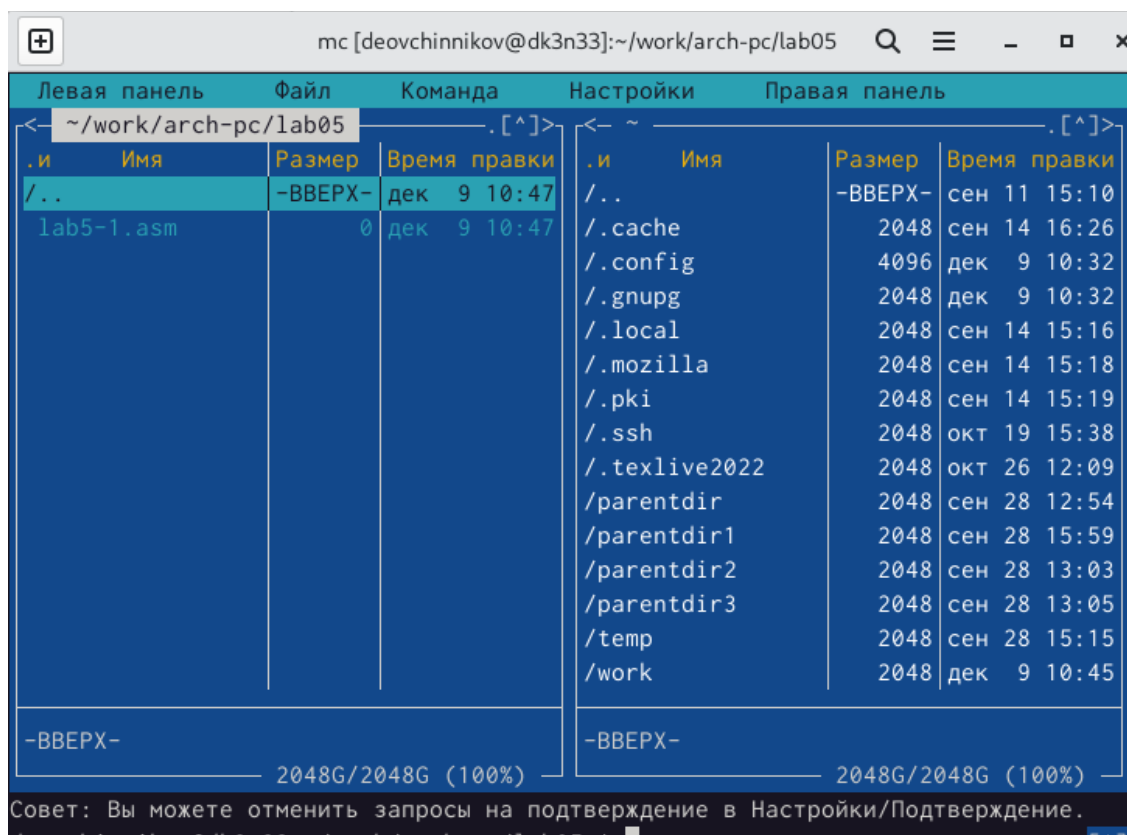


Рис. 3: Создание файла lab5-1.asm

- С помощью функциональной клавиши F4 откроем файл lab5-1.asm.

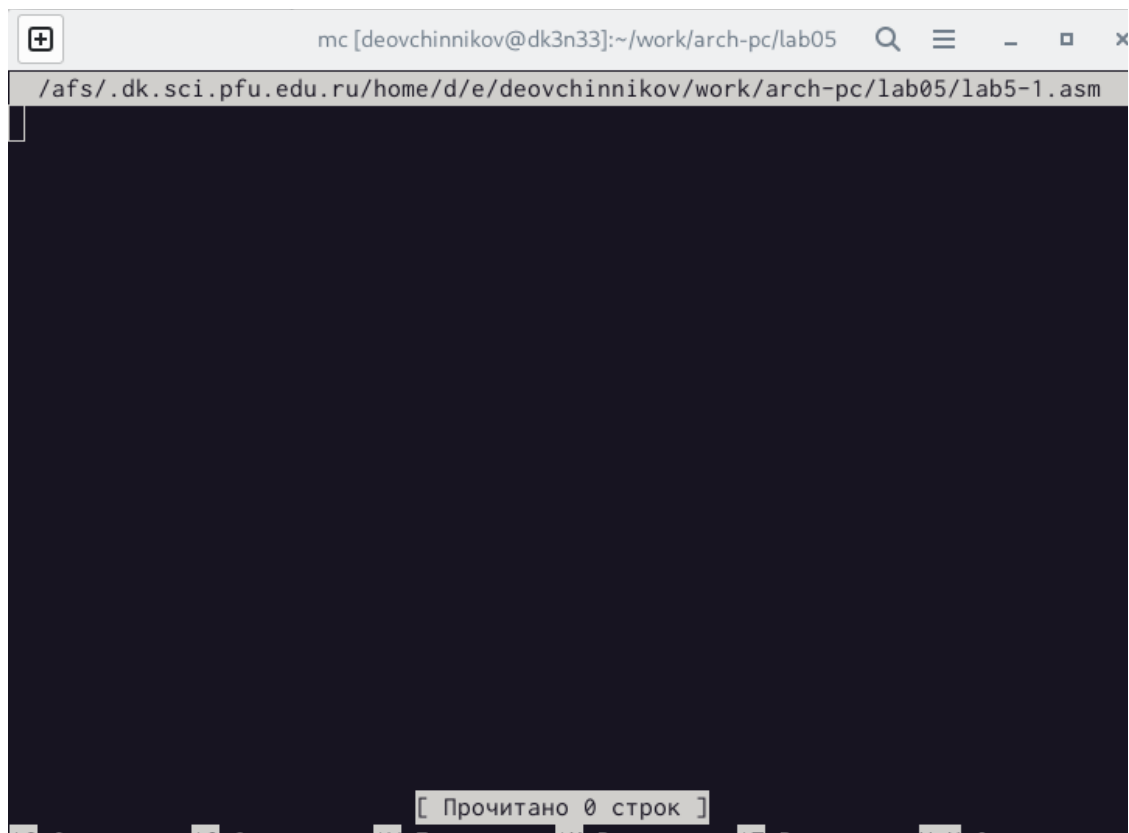
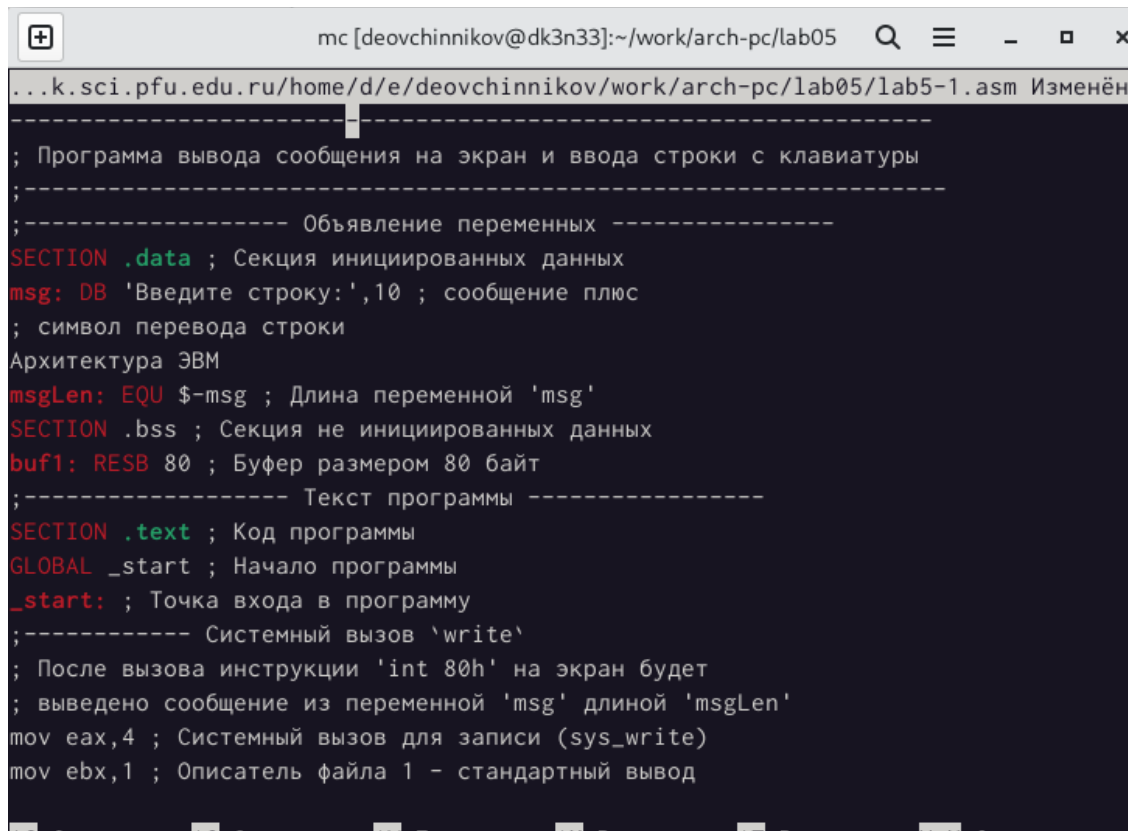


Рис. 4: Открытие файл lab5-1.asm

5. Введём текст программы, сохраним изменения и закроем файл.

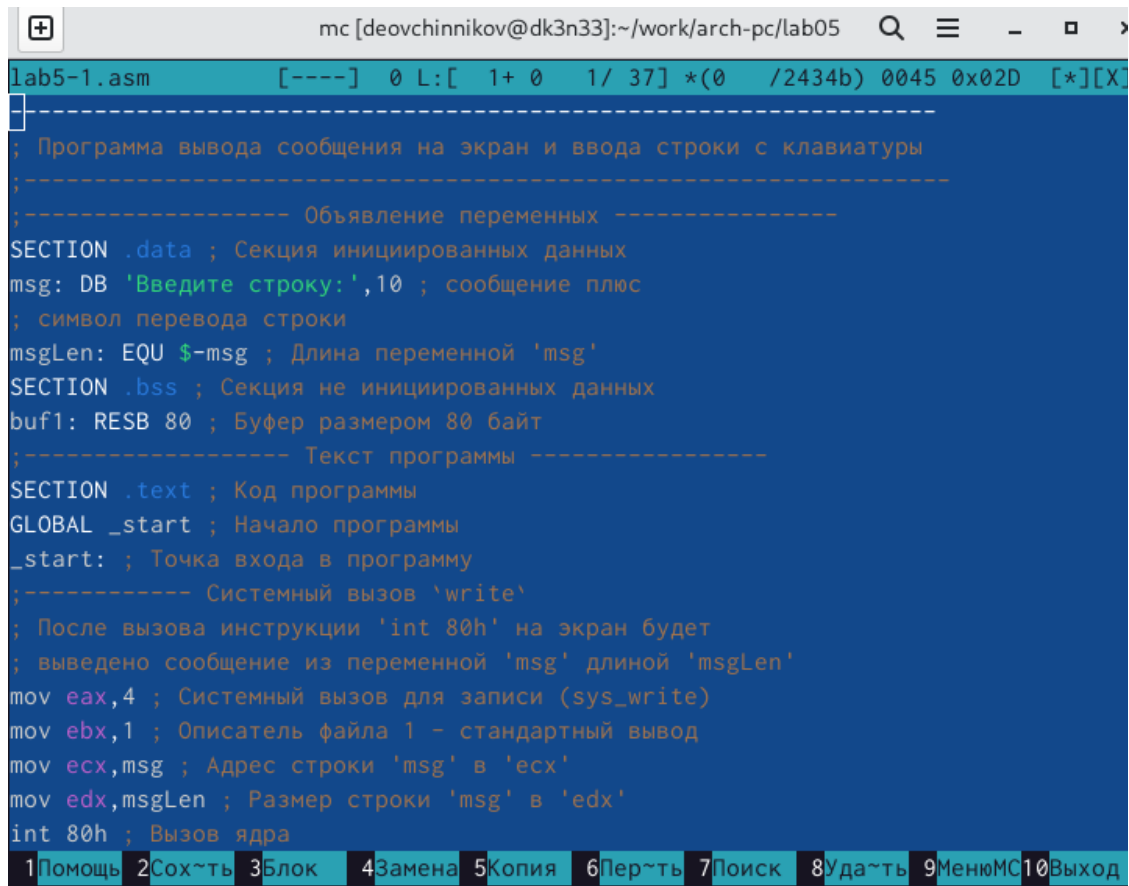


```
mc [deovchinnikov@dk3n33]:~/work/arch-pc/lab05
...k.sci.pfu.edu.ru/home/d/e/deovchinnikov/work/arch-pc/lab05/lab5-1.asm Изменён

; 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
Архитектура ЭВМ
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
```

Рис. 5: Ввод текста, сохранение изменений и закрытие файла

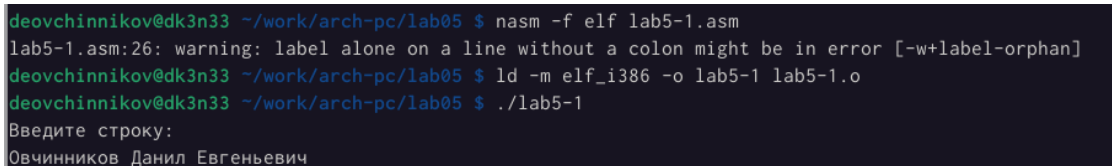
6. С помощью функциональной клавиши F3 откроем файл lab5-1.asm для просмотра.



```
lab5-1.asm [----] 0 L:[ 1+ 0 1/ 37] *(0 /2434b) 0045 0x02D [*][X]
; Программа вывода сообщения на экран и ввода строки с клавиатуры
;
;----- Объявление переменных -----
SECTION .data ; Секция иницированных данных
msg: DB 'Введите строку:',10 ; сообщение плюс
; символ перевода строки
msgLen: EQU $-msg ; Длина переменной 'msg'
SECTION .bss ; Секция не иницированных 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 2Сохранить 3Блок 4Замена 5Копия 6Перейти 7Поиск 8Удалить 9МенюМС10Выход
```

Рис. 6: откроем файл lab5-1.asm для просмотра

7. Оттранслируем текст программы lab5-1.asm в объектный файл. Выполним компоновку объектного файла и запустим получившийся исполняемый файл.



```
deovchinnikov@dk3n33 ~/work/arch-pc/lab05 $ nasm -f elf lab5-1.asm
lab5-1.asm:26: warning: label alone on a line without a colon might be in error [-wlabel-orphan]
deovchinnikov@dk3n33 ~/work/arch-pc/lab05 $ ld -m elf_i386 -o lab5-1 lab5-1.o
deovchinnikov@dk3n33 ~/work/arch-pc/lab05 $ ./lab5-1
Введите строку:
Овчинников Данил Евгеньевич
```

Рис. 7: Оттранслируем текст, выполним компоновку и запустим файл

8. Скачаем файл in_out.asm со страницы курса в ТУИС.
9. Скопируем файл in_out.asm в каталог с файлом lab5-1.asm

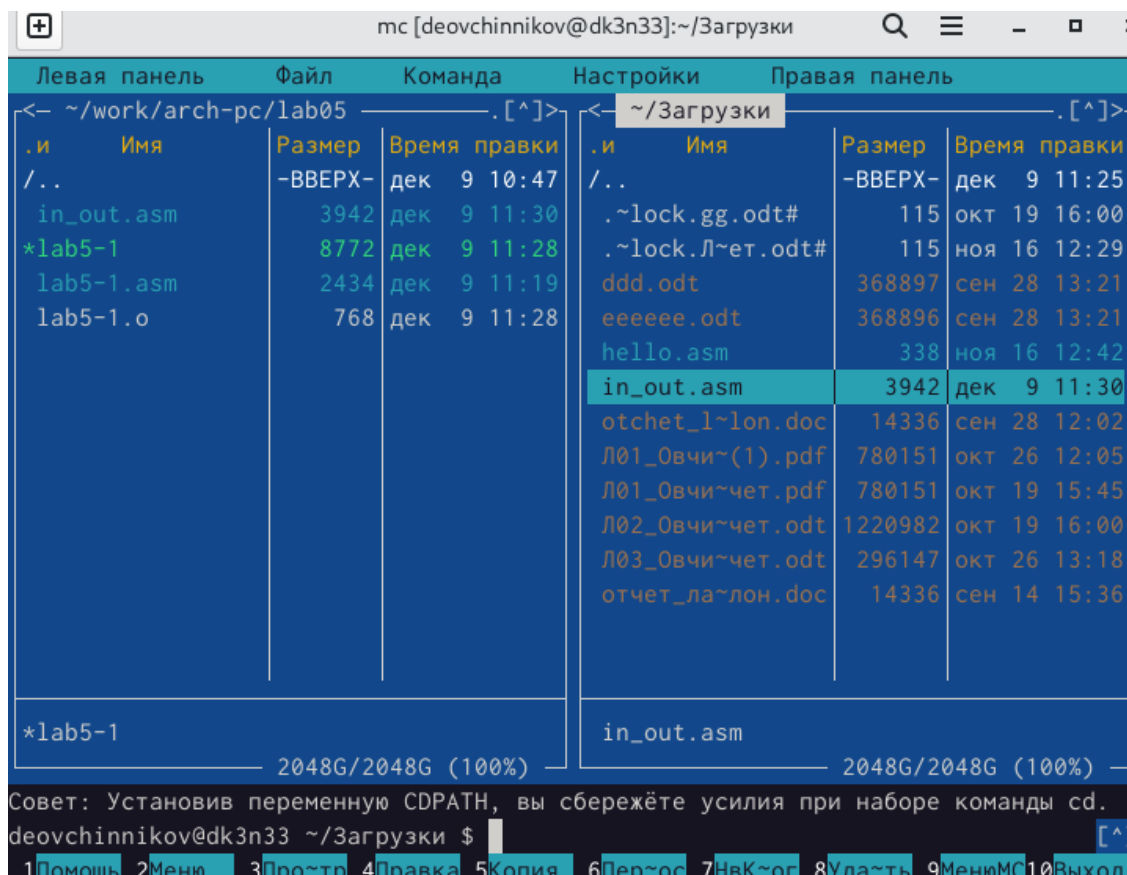


Рис. 8: Скопирование файла

- С помощью функциональной клавиши F6 создадим копию файла lab5-1.asm с именем lab5-2.asm.

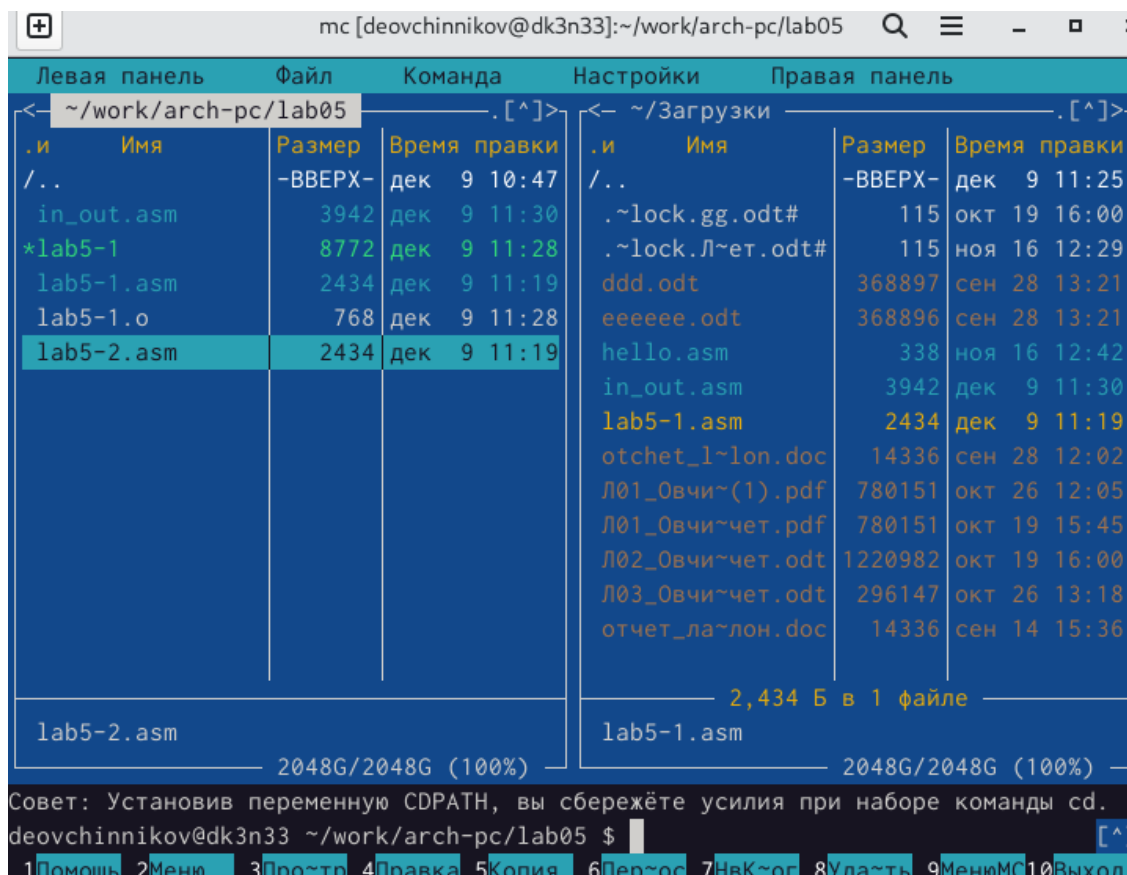
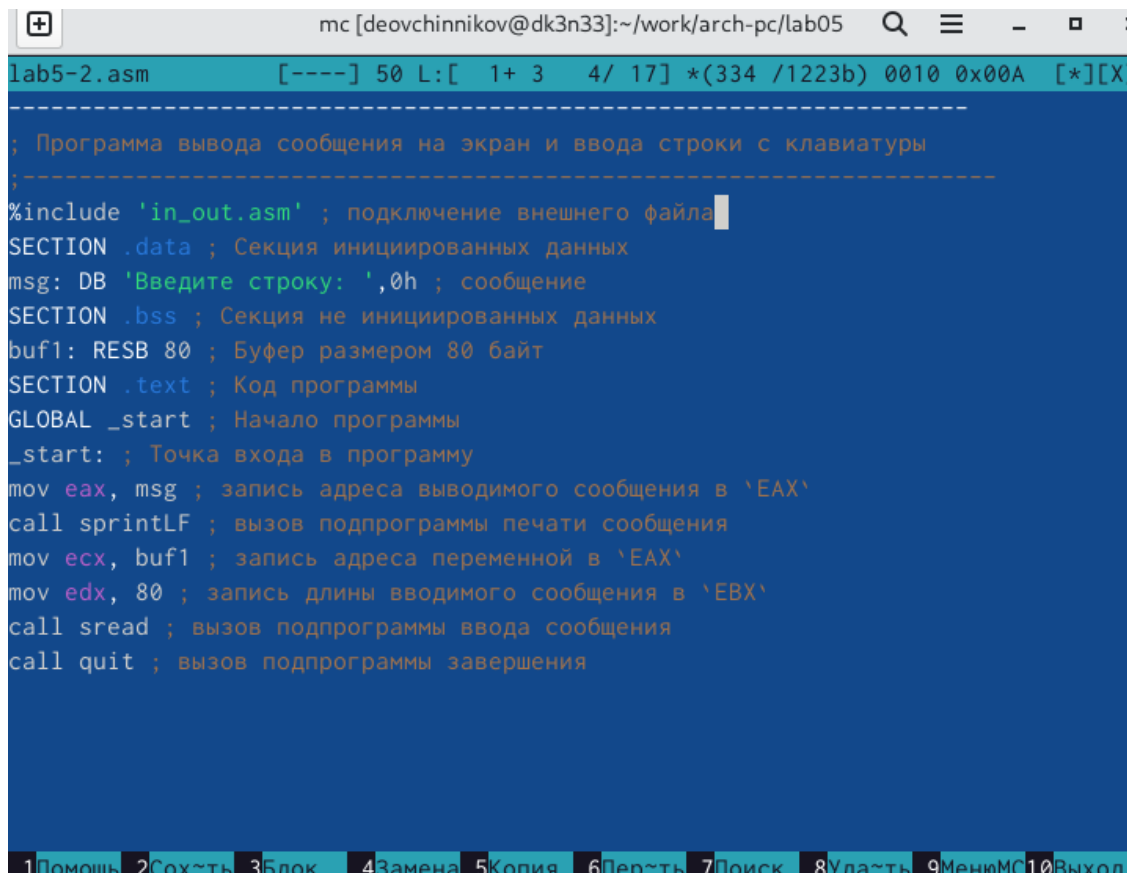


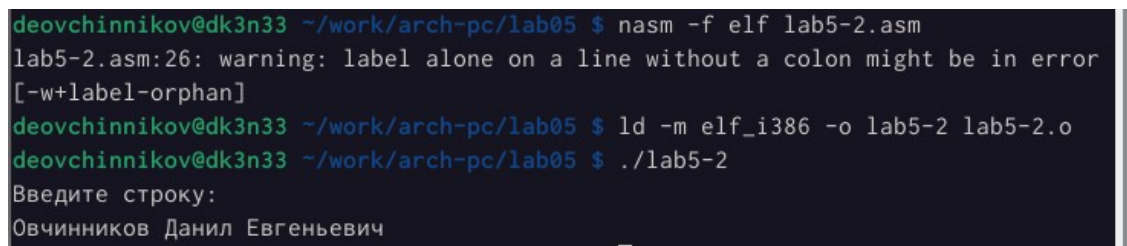
Рис. 9: Создание копии файла

- Исправим текст программы в файле lab5-2.asm с использованием подпрограмм из внешнего файла in_out.asm, создадим исполняемый файл и проверим его работу.



```
lab5-2.asm [----] 50 L:[ 1+ 3 4/ 17] *(334 /1223b) 0010 0x00A [*][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 sprintLF ; 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
```

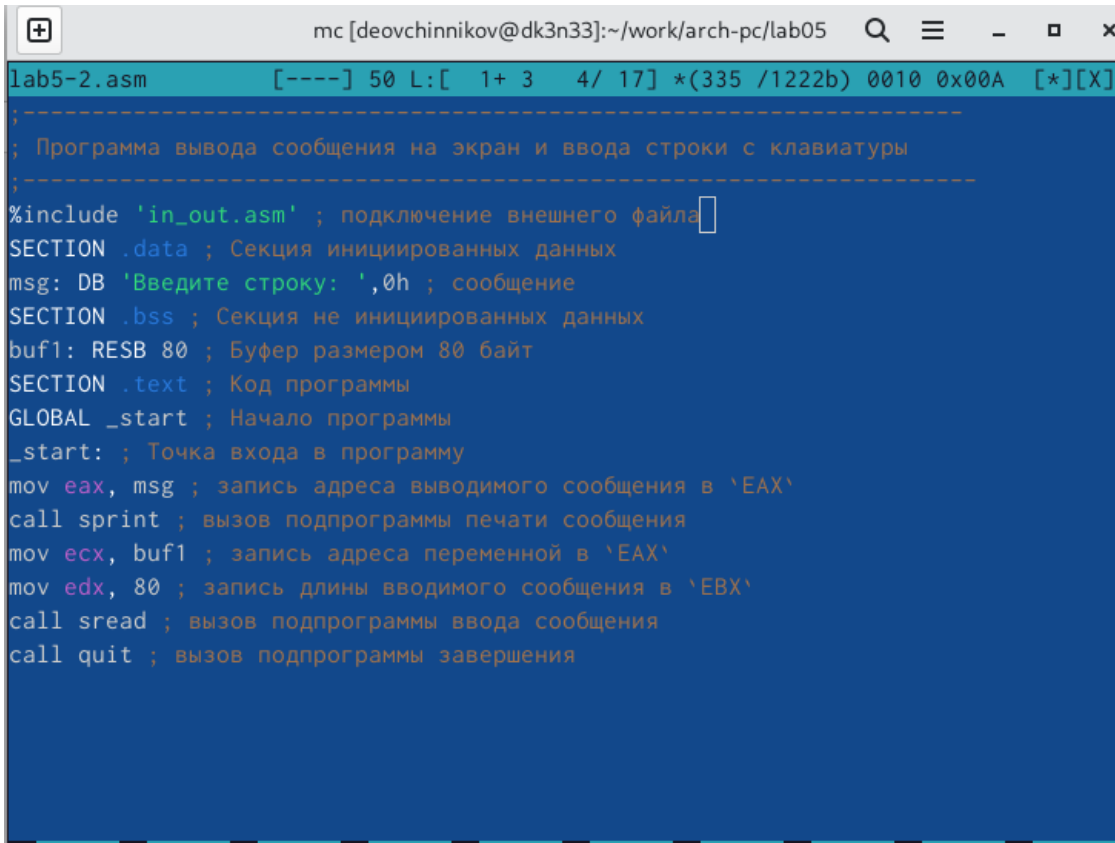
Рис. 10: Исправление текста, создание исполняемого файла и проверка его работы.



```
deovchinnikov@dk3n33 ~/work/arch-pc/lab05 $ nasm -f elf lab5-2.asm
lab5-2.asm:26: warning: label alone on a line without a colon might be in error
[-w+label-orphan]
deovchinnikov@dk3n33 ~/work/arch-pc/lab05 $ ld -m elf_i386 -o lab5-2 lab5-2.o
deovchinnikov@dk3n33 ~/work/arch-pc/lab05 $ ./lab5-2
Введите строку:
Овчинников Данил Евгеньевич
```

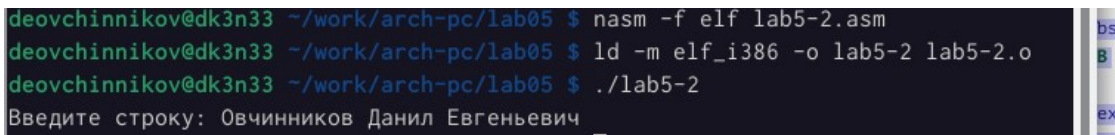
Рис. 11: Исправление текста, создание исполняемого файла и проверка его работы.

12. В файле lab5-2.asm заменим подпрограмму sprintLF на sprint и создадим исполняемый файл и проверьте его работу.



```
lab5-2.asm      [----] 50 L:[ 1+ 3  4/ 17] *(335 /1222b) 0010 0x00A  [*][X]
;
; 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
```

Рис. 12: Заменяем подпрограмму, создадим исполняемый файл и проверим его работу



```
deovchinnikov@dk3n33 ~/work/arch-pc/lab05 $ nasm -f elf lab5-2.asm
deovchinnikov@dk3n33 ~/work/arch-pc/lab05 $ ld -m elf_i386 -o lab5-2 lab5-2.o
deovchinnikov@dk3n33 ~/work/arch-pc/lab05 $ ./lab5-2
Введите строку: Овчинников Данил Евгеньевич
```

Рис. 13: Заменяем подпрограмму, создадим исполняемый файл и проверим его работу

13. Создадим копию файла lab5-1.asm и внесём изменения в программу

./..	-BBEPX-	дек 9 10:47	./..	-BBEPX-	сен 11 15:10
in_out.asm	3942	дек 9 11:30	/.cache	2048	сен 14 16:26
*lab5-1	8772	дек 9 11:28	/.config	4096	дек 9 10:32
lab5-1.asm	2434	дек 9 11:19	/.gnupg	2048	дек 9 10:32
lab5-1.o	768	дек 9 11:28	/.local	2048	сен 14 15:16
lab5-1copy.asm	2434	дек 9 11:19	/.mozilla	2048	сен 14 15:18
*lab5-2	9092	дек 9 11:50	/.pki	2048	сен 14 15:19
lab5-2.asm	1222	дек 9 11:50	/.ssh	2048	окт 19 15:38
lab5-2.o	1312	дек 9 11:50	/.texlive2022	2048	окт 26 12:09
			/parentdir	2048	сен 28 12:54
			/parentdir1	2048	сен 28 15:59
			/parentdir2	2048	сен 28 13:03
			/parentdir3	2048	сен 28 13:05
			/temp	2048	сен 28 15:15
			/work	2048	дек 9 10:45
-BBEPX-			-BBEPX-		
2048G/2048G (100%)			2048G/2048G (100%)		
овет: Требуется вставить литерал? Нажмите Control-q и литерал.					
eovchinnikov@dk3n33 ~/work/arch-pc/lab05 \$					
[^]					
1Помощь 2Меню 3Про~тр 4Правка 5Копия 6Пер~ос 7НвК~ог 8Уда~ть 9МенюМС10Выход					

Рис. 14: Создание копии и внос изменений

```
lab5-1copy.asm  [----]  0 L:[ 1+ 0 1/ 36] *(0 /2434b) 0059 0x03B [*][X]
;
; Программа вывода сообщения на экран и ввода строки с клавиатуры
;
;----- Объявление переменных -----
SECTION .data ; Секция иницированных данных
msg: DB 'Введите строку:',10 ; сообщение плюс
; символ перевода строки
msgLen: EQU $-msg ; Длина переменной 'msg'
SECTION .bss ; Секция не иницированных 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 2Сох~ть 3Блок 4Замена 5Копия 6Пер~ть 7Поиск 8Уда~ть 9МенюМС10Выход
```

Рис. 15: Создание копии и внос изменений

14. Получим исполняемый файл и проверим его работу

```
deovchinnikov@dk3n33 ~/work/arch-pc/lab05 $ nasm -f elf lab5-1copy.asm
lab5-1copy.asm:26: warning: label alone on a line without a colon might be in error [-w+lab
han]
deovchinnikov@dk3n33 ~/work/arch-pc/lab05 $ ld -m elf_i386 -o lab5-1 lab5-1.o
deovchinnikov@dk3n33 ~/work/arch-pc/lab05 $ ld -m elf_i386 -o lab5-1copy lab5-1copy.o
deovchinnikov@dk3n33 ~/work/arch-pc/lab05 $ ./lab5-1copy
Введите строку:
Овчинников
```

Рис. 16: Проверка работы файла

15. Создадим копию файла lab5-2.asm. Исправим текст программы с использованием подпрограмм из внешнего файла in_out.asm.

~ /work/arch-pc/lab05				~ .[^]>			
.и	Имя	Размер	Время правки	.и	Имя	Размер	Время правки
./..		-ВВЕРХ-	дек 9 10:47	./..		-ВВЕРХ-	сен 11 15:10
in_out.asm		3942	дек 9 11:30	./.cache		2048	сен 14 16:26
*lab5-1		8772	дек 9 12:01	./.config		4096	дек 9 10:32
lab5-1.asm		2434	дек 9 11:19	./.gnupg		2048	дек 9 10:32
lab5-1.o		768	дек 9 11:28	./.local		2048	сен 14 15:16
*lab5-1copy		8776	дек 9 12:01	./.mozilla		2048	сен 14 15:18
lab5-1copy.asm		2434	дек 9 11:19	./.pki		2048	сен 14 15:19
lab5-1copy.o		784	дек 9 12:01	./.ssh		2048	окт 19 15:38
*lab5-2		9092	дек 9 11:50	./.texlive2022		2048	окт 26 12:09
lab5-2.asm		1222	дек 9 11:50	/parentdir		2048	сен 28 12:54
lab5-2.o		1312	дек 9 12:00	/parentdir1		2048	сен 28 15:59
lab5-2copy.asm		1222	дек 9 11:50	/parentdir2		2048	сен 28 13:03
				/parentdir3		2048	сен 28 13:05
				/temp		2048	сен 28 15:15
				/work		2048	дек 9 10:45
				/Видео		2048	сен 14 15:16
				/Документы		2048	сен 14 15:16
-ВВЕРХ-				-ВВЕРХ-			
2048G/2048G (100%)				2048G/2048G (100%)			
Совет: Домашняя страница Midnight Commander: http://www.midnight-commander.org/							
deovchinnikov@dk3n33 ~/work/arch-pc/lab05 \$							

Рис. 17: Создание копии и внос изменений

```
;-----  
; 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 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
```

[Прочитано 17 строк]

^G Справка ^O Записать ^W Поиск ^K Вырезать ^T Выполнить M-U Отмена

Рис. 18: Создание копии и внос изменений

Создадим исполняемый файл и проверим его работу.

```
deovchinnikov@dk3n33: ~/work/arch-pc/lab05  
deovchinnikov@dk3n33 ~ $ cd ~/work/arch-pc/lab05  
deovchinnikov@dk3n33 ~/work/arch-pc/lab05 $ nasm -f elf lab5-2copy.asm  
deovchinnikov@dk3n33 ~/work/arch-pc/lab05 $ ld -m elf_i386 -o lab5-2copy lab5-2copy.o  
deovchinnikov@dk3n33 ~/work/arch-pc/lab05 $ ./lab5-2copy  
Введите строку: Овчинников
```

Рис. 19: Проверка работы файла

3 Выводы

В ходе выполнения лабораторной работы я приобрёл практические навыки работы в Midnight Commander и освоил инструкции языка ассемблера `mov` и `int`.