

**Отчёт по лабораторной работе №5.  
Основы работы с Midnight Commander  
(mc). Структура программы на языке  
ассемблера NASM. Системные вызовы в  
ОС GNU Linux**

**Архитектура вычислительных систем**

Гандич Дарья Владимировна. НБИбд-02-22.

# Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение лабораторной работы	7
4	Вывод	15

## Список иллюстраций

3.1	Создание файла . . . . .	8
3.2	Листинг 6.1 . . . . .	9
3.3	Трансляция текста . . . . .	10
3.4	Копирование файла in_out.asm . . . . .	10

## **Список таблиц**

# 1 Цель работы

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

## 2 Задание

1. Создайте копию файла lab6-1.asm. Внесите 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 введённую строку на экран.
2. Получите исполняемый файл и проверьте его работу. На приглашение ввести строку введите свою фамилию
3. Создайте копию файла lab6-2.asm. Исправьте текст программы с использование подпрограмм из внешнего файла in\_out.asm, так чтобы она работала по следующему алгоритму: • вывести приглашение типа “Введите строку:”; • ввести строку с клавиатуры; • вывести введённую строку на экран.

### 3 Выполнение лабораторной работы

1. Откроем Midnight Commander и создадим каталог lab05 (сначала создала каталог в lab04, так как пункты 2 и 3 были друг за другом. потрм перенесла).

Левая панель				Правая панель			
Файл				Настройки			
[...]				[...]			
Имя	Размер	Время	Правки	Имя	Размер	Время	Правки
..	-ВВЕРХ-	ноя	2 12:26	..	-ВВЕРХ-	окт	12 15:11
/.git	2048	ноя	23 13:58	/.cache	2048	сен	7 11:29
/config	2048	ноя	2 12:26	/.config	4096	дек	7 10:44
/labs	2048	ноя	2 12:27	/.emacs.d	2048	сен	28 10:55
/template	2048	ноя	2 12:27	/.fltk	2048	ноя	2 13:43
.gitattributes	1765	ноя	2 12:26	/.gnupg	2048	дек	7 10:37
.gitignore	4637	ноя	2 12:26	/.kde4	2048	ноя	9 14:30
.gitmodules	278	ноя	2 12:26	/.local	2048	сен	7 11:17
CHANGELOG.md	2120	ноя	2 12:26	/.mozilla	2048	сен	7 11:17
COURSE	8	ноя	2 12:31	/.ssh	2048	ноя	2 12:26
LICENSE	18657	ноя	2 12:26	/.texlive2022	2048	ноя	2 12:34
Makefile	815	ноя	2 12:26	/CC+	2048	ноя	9 11:54
README.en.md	152	ноя	2 12:26	/tmp	2048	сен	28 10:55
README.git-flow.md	5653	ноя	2 12:26	/work	2048	ноя	2 12:23
README.md	4477	ноя	2 12:26	/Видео	2048	сен	7 11:17
prepare	0	ноя	2 12:27	/Документы	2048	сен	7 11:17
report.md	5128	ноя	2 13:50	/Загрузки	2048	ноя	2 15:28
				/Изображения	2048	ноя	16 11:39
				/Музыка	2048	сен	7 11:17
				/Общедоступные	2048	сен	7 11:17
				/Рабочий стол	2048	ноя	9 14:31
				/Шаблоны	2048	сен	7 11:17
				.xauthority	561	дек	7 10:37
				.bash_history	6107	ноя	23 14:00
				.bash_profile	245	сен	3 2012
				.bashrc	124	апр	23 2012
				.dmrc	32	ноя	2 14:41
				.gitconfig	140	ноя	2 12:17
				.gtkrc-2.0	264	ноя	9 14:36
				.profile	536	сен	12 2016
				.xsession-errors	0	дек	7 10:37
				.xsession-errors.old	0	ноя	23 12:49
				kol	10204	сен	28 11:27
				kol.cpp	146	сен	28 11:27
				kol.o	3168	сен	28 11:27
				архив_ЛР04_Гандич.zip	901955	ноя	23 13:53

Левая панель	Файл	Команда	Настройки	Прав.
<...3/Архитектура компьютера/arch-pc/labs/lab04 -.[^]>				
Имя	Размер	Время правки		
/..	-ВВЕРХ-	ноя 2 12:27		
/lab05	2048	дек 7 10:48		
/presentation	2048	ноя 2 12:26		
/report	2048	ноя 23 13:53		
COURSE	8	ноя 23 13:10		
*hello	8668	ноя 16 11:10		
hello.asm	849	ноя 16 11:04		
hello.o	656	ноя 16 11:05		
*lab04	8676	ноя 16 11:25		
lab04.asm	871	ноя 16 11:22		
lab04.o	672	ноя 16 11:22		
list.lst	1663	ноя 16 11:24		
*main	9060	ноя 16 11:10		
obj.o	1568	ноя 16 11:24		

- Через командную строку создаем файл lab5-1.asm (с помощью команды touch).

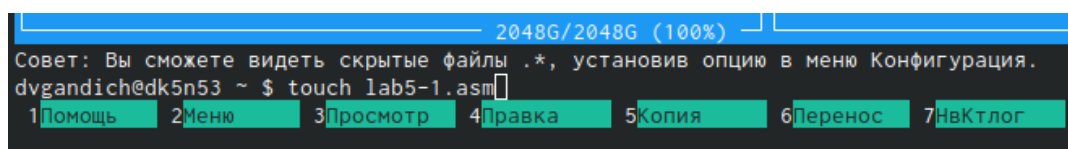


Рис. 3.1: Создание файла



3. Проверяем, что файл создан, переходим в него и вводим текст программы из листинга 6.1 из материалов.

```
-----  
; Программа вывода сообщения на экран и ввода строки с клавиатуры  
-----  
  
;----- Объявление переменных -----  
SECTION .data  
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)
```

Рис. 3.2: Листинг 6.1

4. Оттранслируем текст программы lab5-1.asm в объектный файл через строку

ввода, программа должна вывести “Введите строку:”.

```
dvgandichedk3n52 ~ $ cd ~/work/study/2022-2023/"Архитектура компьютера"/arch-pc/labs/lab05
dvgandichedk3n52 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab05 $ nasm -f elf lab5-1.asm
lab5-1.asm:30: warning: label alone on a line without a colon might be in error [-w+label-orphan]
dvgandichedk3n52 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab05 $ ld -m elf_i386 -o lab5-1 lab5-1.o
dvgandichedk3n52 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab05 $ ./lab5-1
Введите строку:
Gandich Darya
dvgandichedk3n52 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab05 $
```

Рис. 3.3: Трансляция текста

5. Скачиваем файл in\_out.asm и копируем его в каталог с файлом lab5-1.asm

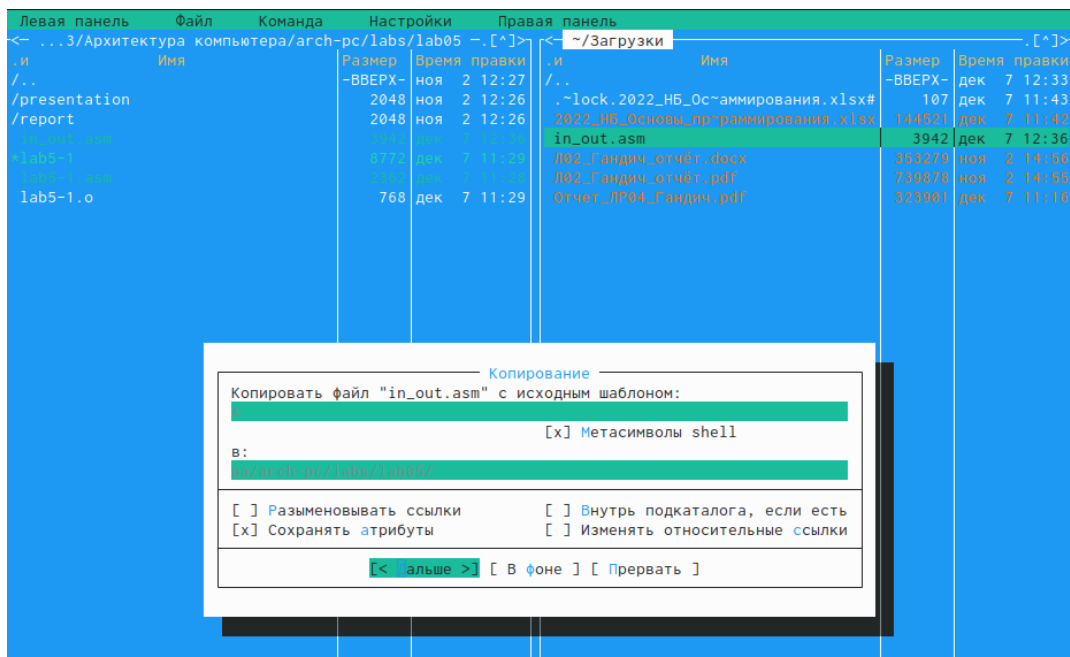


Рис. 3.4: Копирование файла in\_out.asm

6. Создаем копию файла lab5-1.asm с именем lab5-2.asm и перепишем текст программы из листинга 6.2.

...3/Архитектура компьютера/arch-pc/labs/lab05			
Имя	Размер	Время правки	
/..	-ВВЕРХ-	ноя	2 12:27
/presentation	2048	ноя	2 12:26
/report	2048	ноя	2 12:26
in_out.asm	3942	дек	7 12:36
*lab5-1	8772	дек	7 11:29
lab5-1.o	768	дек	7 11:29
lab5-2.asm	2362	дек	7 11:28

```

;-----
; 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

```

7. Заменим в тексте подпрограмму sprintf на printf, создадим исполняемые файлы и заметим разницу: при подпрограмме printf строка ввода сносится вниз, а при printf остается на той же строке, где и находится условие

```

mov eax, msg ; запись адреса выводимого сообщения в 'EAX'
call printf ; вызов подпрограммы печати сообщения

```

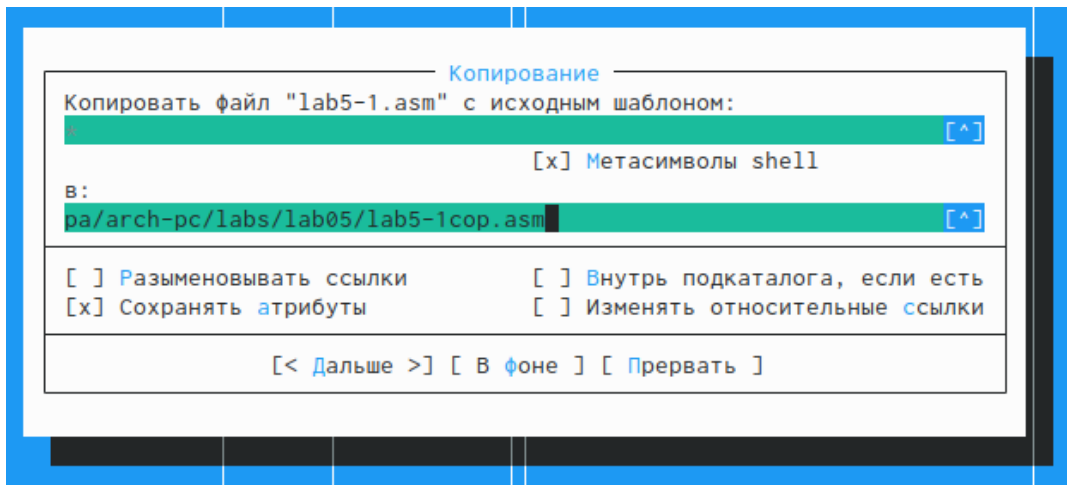
```

dvgandich@dk2n25 ~ $ cd ~/work/study/2022-2023/"Архитектура компьютера"/arch-pc/labs/lab05
dvgandich@dk2n25 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab05 $ nasm -f elf lab5-2.asm
dvgandich@dk2n25 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab05 $ ld -m elf_i386 -o lab5-2 lab5-2.o
dvgandich@dk2n25 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab05 $ ./lab5-2
Введите строку:
Gandich Darya
dvgandich@dk2n25 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab05 $ nasm -f elf lab5-2.asm
dvgandich@dk2n25 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab05 $ ld -m elf_i386 -o lab5-2 lab5-2.o
dvgandich@dk2n25 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab05 $ ./lab5-2
Введите строку: Gandich Darya

```

Задание для самостоятельной работы:

1. Создаем копию файла lab5-1.asm, вносим изменения в программу, так чтобы она работала по алгоритму из условия.



```

;-----
; Программа вывода сообщения на экран и ввода строки с клавиатуры
;-----

;----- Объявление переменных -----
SECTION .data
msg: DB 'Введите строку:',10      ; сообщение плюс
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

;----- 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

mov eax, 4
mov ebx, 1
mov ecx, buf1
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

dvgandich@dk5n53 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab05 $ nasm -f elf lab5-1cop.asm
lab5-1cop.asm:30: warning: label alone on a line without a colon might be in error [-w+label-orphan]
dvgandich@dk5n53 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab05 $ ld -m elf_i386 -o lab5-1cop lab5-1cop.o
dvgandich@dk5n53 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab05 $ ./lab5-1cop
Введите строку:
Gandich Darya
Gandich Darya
dvgandich@dk5n53 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab05 $ █

```

2. Теперь сделаем все то же самое с файлом lab5-2.asm

lab5-1cop.asm	2424	дек	7 14:07
lab5-1cop.o	800	дек	9 12:03
*lab5-2	9092	дек	7 13:44
lab5-2.asm	1265	дек	7 13:45
lab5-2.o	1328	дек	7 13:44
*lab5-2cop	9092	дек	7 14:09
lab5-2cop.asm	1296	дек	7 13:43
lab5-2cop.o	1328	дек	7 14:08

```

;-----
; 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

mov eax, buf1 ;
call sprint ;

call quit                      ; вызов подпрограммы завершения

dvgandich@dk2n25 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab05 $ nasm -f elf lab5-2cop.asm
dvgandich@dk2n25 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab05 $ ld -m elf_i386 -o lab5-2cop lab5-2cop.o
dvgandich@dk2n25 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab05 $ ./lab5-2cop
Введите строку: Gandich
Gandich

```

## 4 Вывод

Мы научились работать в тс, приобрели практические навыки в нем, освоили инструкции языка ассемблера.