

**Лабораторная работа №5. Основы
работы Midnight Commander. Структура
программы на языке ассемблера.
Системные вызовы в ОС Linux отчёта по
лабораторной работе**

Отчёт по лабораторной работе

Павленко Сергей

Содержание

1	Цель работы	5
2	Теоретическое введение	6
3	Выполнение лабораторной работы	7
4	Выводы	12
5	Задание для самостоятельной работы	13
6	Выводы	14
	Список литературы	15

Список иллюстраций

Список таблиц

1 Цель работы

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

2 Теоретическое введение

Основы работы с Midnight Commander Midnight Commander (или просто mc) — это программа, которая позволяет просматривать структуру каталогов и выполнять основные операции по управлению файловой системой, т.е. mc является файловым менеджером. Midnight Commander позволяет сделать работу с файлами более удобной и наглядной. Для активации оболочки Midnight Commander достаточно ввести в командной строке mc и нажать клавишу Enter В Midnight Commander используются функциональные клавиши F1 — F10 , к которым привязаны часто выполняемые операции Дополнительную информацию о Midnight Commander можно получить по команде man mc и на странице проекта

3 Выполнение лабораторной работы

1. Откройте Midnight Commander `user@dk4n31:~$ mc`
2. Пользуясь клавишами `⌘`, `⌘` и `Enter` перейдите в нужный каталог `~/work/arch-`

```
mc [spavlenko@spavlenko]:~/work/study/2023-2024/Архитектура компь...

Left  File      Command  Options  Right
-----
<- ...tudy_2023-2024_arhpc/labs -.[^]>
.n     Name      Size     Modify   time
/..    UP--DIR      4096    Oct 13 20:21
/lab01 4096    Oct 13 20:21
/lab02 4096    Oct 13 20:21
/lab03 4096    Oct 13 20:21
/lab04 4096    Oct 28 05:49
/lab05 4096    Oct 13 20:21
/lab06 4096    Oct 13 20:21
/lab07 4096    Oct 13 20:21
/lab08 4096    Oct 13 20:21
/lab09 4096    Oct 13 20:21
/lab10 4096    Oct 13 20:21
/lab11 4096    Oct 13 20:21
README.md      19    Oct 13 20:21
README.ru.md   40    Oct 13 20:21

UP--DIR      28G / 49G (56%)

Hint: To change directory halfway through typing a command, use M-c (quick cd).
...arhpc/labs$

1Help 2Menu 3View 4Edit 5Copy 6RenMov 7Mkdir 8Delete 9PullDn 10Quit
```

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

```

mc [spavlenko@spavlenko]:~/work/study/2023-2024/Архитектура компь...
Left File Command Options Right
<- ...4 arhpc/labs/lab05/report .[^]>
.n Name Size Modify time .n Name Size Modify time
/.. UP--DIR Oct 13 20:21 /.. UP--DIR Oct 13 06:17
/bib 4096 Oct 13 20:21 /.cache 4096 Nov 10 10:46
/image 4096 Oct 13 20:21 /.config 4096 Nov 10 10:46
/pandoc 4096 Oct 13 20:21 /.gnupg 4096 Oct 28 22:22
Makefile 1064 Oct 13 20:21 /.local 4096 Oct 13 08:37
report.md 5647 Oct 13 20:21 /.ssh 4096 Oct 13 20:20
/.texlive2023 4096 Oct 14 12:02
/Desktop 4096 Oct 13 09:51
/Documents 4096 Oct 13 08:37
/Downloads 4096 Oct 28 22:26
/Music 4096 Oct 13 08:37
/Pictures 4096 Oct 14 17:35
/Public 4096 Oct 13 08:37
/Templates 4096 Oct 13 08:37
/Videos 4096 Oct 13 08:37

UP--DIR 28G / 49G (56%) UP--DIR 28G / 49G (56%)
Hint: M-! will allow you to execute programs and see the output in the viewer.
4_arhpc/labs/lab05/report$
1Help 2Menu 3View 4Edit 5Copy 6RenMov 7Mkdir 8Delete 9PullDn 10Quit

```

- С помощью функциональной клавиши F4 откройте файл lab5-1.asm для редактирования во встроенном редакторе. Как правило в качестве встроенного редактора Midnight Commander используется редакторы nano

```

Left File Command Options Right
<- ...4 arhpc/labs/lab05/report .[^]>
.n Name Size Modify time .n Name Size Modify time
/.. UP--DIR Oct 13 20:21 /.. UP--DIR Oct 13 06:17
/bib 4096 Oct 13 20:21 /.cache 4096 Nov 10 10:46
/image 4096 Oct 13 20:21 /.config 4096 Nov 10 10:46
/pandoc 4096 Oct 13 20:21 /.gnupg 4096 Oct 28 22:22
Makefile 1064 Oct 13 20:21 /.local 4096 Oct 13 08:37
lab5-1.asm 0 Nov 10 10:58 /.ssh 4096 Oct 13 20:20
report.md 5647 Oct 13 20:21 /.texlive2023 4096 Oct 14 12:02
/Desktop 4096 Oct 13 09:51
/Documents 4096 Oct 13 08:37
/Downloads 4096 Oct 28 22:26
/Music 4096 Oct 13 08:37
/Pictures 4096 Oct 14 17:35
/Public 4096 Oct 13 08:37
/Templates 4096 Oct 13 08:37
/Videos 4096 Oct 13 08:37

UP--DIR 28G / 49G (56%) UP--DIR 28G / 49G (56%)
Trash 28G / 49G (56%)
Hint: M-! will allow you to execute programs and see the output in the viewer.
4_arhpc/labs/lab05/report$
1Help 2Menu 3View 4Edit 5Copy 6RenMov 7Mkdir 8Delete 9PullDn 10Quit

```

- Введите текст программы из листинга 5.1 (можно без комментариев),

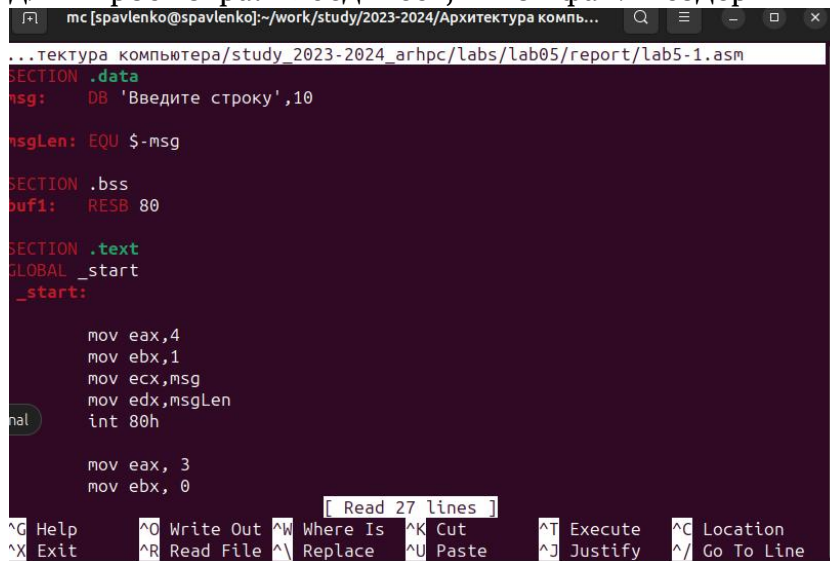

```
spavlenko@spavlenko:~$ mc
4_arhpc/labs/lab05/report$ touch lab5-1.asm

Select an editor. To change later, run 'select-editor'.
 1. /bin/nano          <---- easiest
 2. /usr/bin/mcedit
 3. /usr/bin/vim.tiny
 4. /bin/ed

Choose 1-4 [1]: 1
```

сохраните изменения и закройте файл

6. С помощью функциональной клавиши F3 откройте файл lab5-1.asm для просмотра. Убедитесь, что файл содержит текст программы.



```
..текстура компьютера/study_2023-2024_arhpc/labs/lab05/report/lab5-1.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

[ Read 27 lines ]
^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute   ^C Location
^X Exit      ^R Read File ^N Replace   ^U Paste     ^J Justify   ^_ Go To Line
```

7. Оттранслируйте текст программы lab5-1.asm в объектный файл. Выполните компоновку объектного файла и запустите получившийся исполняемый файл. Программа выводит строку 'Введите строку:' и ожидает ввода с клавиатуры. На запрос введите Ваши ФИО. user@dk4n31:~\$ nasm -f elf lab5-1.asm user@dk4n31:~\$ ld -m elf_i386 -o lab5-1 lab5-1.o user@dk4n31:~\$./lab5-1 Введите строку: Имя пользователя user@dk4n31:~\$

```
mc [spavlenko@spavlenko]:~/work/study/2023-2024/Архитектура компь...
/home/spavlenko/work/study/2023-2024/Архитектура компь... 254/295 86%
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

1help 2UnWrap 3Quit 4Hex 5Goto 6 7Search 8Raw 9Format10Quit
```

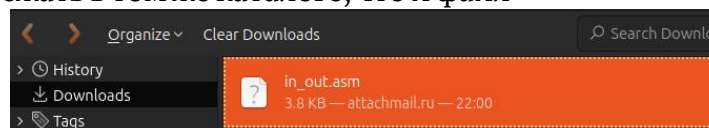
```
spavlenko@spavlenko:~$ mc
4_arhpc/labs/lab05/report$ nasm -f elf lab5-1.asm
4_arhpc/labs/lab05/report$ ld -m elf_i386 lab5-1.o -o lab5-1
4_arhpc/labs/lab05/report$ ./lab5-1
Введите строку
Имя пользователя
```

8. Скачайте файл in_out.asm со страницы курса в ТУИС.

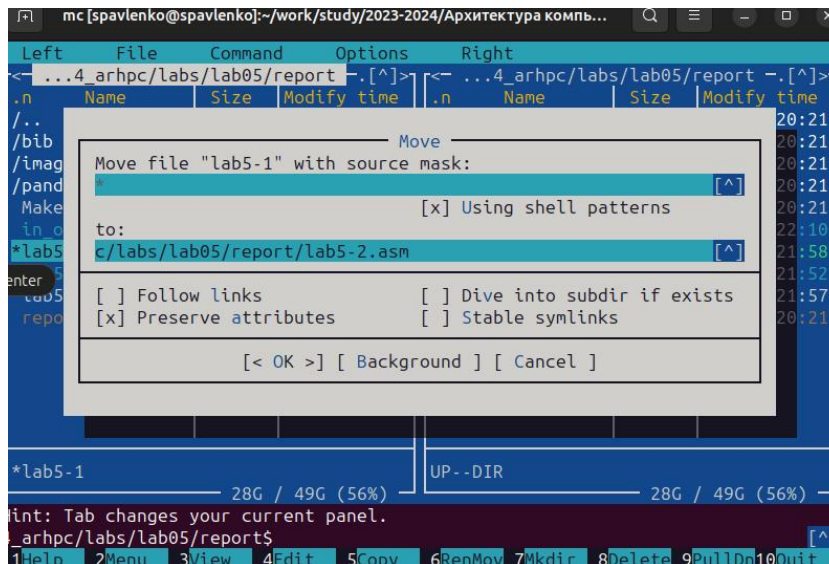
9. Подключаемый файл in_out.asm должен лежать в том же каталоге, что и файл

с программой, в которой он используется.

```
spavlenko@spavlenko:~/Downloads$ cp in_out.asm ~/work/study/2023-2024/Архитектура компь...
a\ компьютера/study_2023-2024_arhpc/labs/lab05/report/
spavlenko@spavlenko:~/Downloads$
```



10. С помощью функциональной клавиши F6 создайте копию файла lab5-1.asm с именем lab5-2.asm. Выделите файл lab5-1.asm, нажмите клавишу F6 , введите имя файла lab5-2.asm и нажмите клавишу Enter



11. Исправьте текст программы в файле lab5-2.asm с использованием подпрограмм из внешнего файла in_out.asm (используйте подпрограммы sprintLF, sread и quit) в соответствии с листингом 5.2. Создайте исполняемый файл и проверьте его работу

```
spavlenko@spavlenko:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arhpc/labs/lab05/report$ nasm -f elf lab5-2.asm
spavlenko@spavlenko:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arhpc/labs/lab05/report$ ld -m elf_i386 -o lab5-2 lab5-2.o
```

12. В файле lab5-2.asm замените подпрограмму sprintLF на sprint. Создайте исполняемый файл и проверьте его работу. В чем разница?

```
spavlenko@spavlenko:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arhpc/labs/lab05/report$ ./lab5-2
Введите строку
call sprintLF
spavlenko@spavlenko:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arhpc/labs/lab05/report$
```

А разница в том, что sprint – вывод сообщения на экран, а sprintLF – работает аналогично sprint, но при выводе на экран добавляет к сообщению символ перевода строки;

4 Выводы

Таким образом, узнали, что `mc` позволяет просматривать структуры каталогов и выполнять основные операции по управлению файловой системой, а также мы научились базово пользоваться `mc`, узнали различные структуры программ на языке `NASM`.

5 Задание для самостоятельной работы

1. Создайте копию файла lab5-1.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 введенную строку на экран.

```
spavlenko@spavlenko:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arhpc/labs/lab05/report$ ./lab5-2
Введите строку
call sprint
spavlenko@spavlenko:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arhpc/labs/lab05/report$
```

2. Получите исполняемый файл и проверьте его работу. На приглашение

```
spavlenko@spavlenko:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arhpc/labs/lab05/report$ ./lab5-1
Введите строку:
hello world!
spavlenko@spavlenko:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arhpc/labs/lab05/report$ ld -m elf_i386 -o lab5-1 lab5-1.o
spavlenko@spavlenko:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arhpc/labs/lab05/report$
```

ввести строку введите свою фамилию.

3. Создайте копию файла lab5-2.asm. Исправьте 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

```
spavlenko@spavlenko:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arhpc/labs/lab05/report$ ./lab5-2
Введите строку
Pavlenko
spavlenko@spavlenko:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arhpc/labs/lab05/report$ ld -m elf_i386 -o lab5-2 lab5-2.o
spavlenko@spavlenko:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arhpc/labs/lab05/report$
```

строку на экран.

6 Выводы

В ходе лабораторной работы мы выяснили назначение мс, какие операции с файлами можно выполнять с помощью мс, основные структуры ассемблера, какие компоненты и для чего используются в языке ассемблере, итд.

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.
12. Новожилов О. П. Архитектура ЭВМ и систем. — М. : Юрайт, 2016.
13. Расширенный ассемблер: NASM. — 2021. — URL: <https://www.opennet.ru/docs/RUS/nasm/>.
14. Робачевский А., Немнюгин С., Стесик О. Операционная система UNIX. — 2-е изд. — БХВ- Петербург, 2010. — 656 с. — ISBN 978-5-94157-538-1.
15. 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.
16. Таненбаум Э. Архитектура компьютера. — 6-е изд. — СПб. : Питер, 2013. — 874 с. — (Классика Computer Science).
 17. Таненбаум Э., Бос Х. Современные операционные системы. — 4-е изд. — СПб. : Питер,
 18. — 1120 с. — (Классика Computer Science)