

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра физико-математических и естественных наук

ОТЧЕТ

ПО ЛАБОРАТОРНОЙ РАБОТЕ № 5

дисциплина: Архитектура компьютера

Студент: Попова Алиса Владимировна

Группа: НПИ-03-25

МОСКВА

2025г.

Цель работы

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

Теоретическая часть

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 (рис. 5.1).

В Midnight Commander используются функциональные клавиши F1 — F10 , к которым привязаны часто выполняемые операции (табл. 5.1)

Таблица 5.1. Функциональные клавиши Midnight Commander

Функцио- нальные клавиши	Выполняемое действие
F1	вызов контекстно-зависимой подсказки
F2	вызов меню, созданного пользователем
F3	просмотр файла, на который указывает подсветка в активной панели
F4	вызов встроенного редактора для файла, на который указывает подсветка в активной панели
F5	копирование файла или группы отмеченных файлов из каталога, отображаемого в активной панели, в каталог, отображаемый на второй панели
F6	перенос файла или группы отмеченных файлов из каталога, отображаемого в активной панели, в каталог, отображаемый на второй панели
F7	создание подкаталога в каталоге, отображаемом в активной панели
F8	удаление файла (подкаталога) или группы отмеченных файлов
F9	вызов основного меню программы
F10	выход из программы

Следующие комбинации клавиш облегчают работу с Midnight Commander:

- **Tab** используется для переключения между панелями;
- **↑** и **↓** используется для навигации, **Enter** для входа в каталог или открытия файла (если в файле расширений `mc.ext` заданы правила связи определённых расширений файлов с инструментами их запуска или обработки);
- **Ctrl + u** (или через меню **Команда** > **Переставить панели**) меняет местами содержимое правой и левой панелей;
- **Ctrl + o** (или через меню **Команда** > **Отключить панели**) скрывает или возвращает панели Midnight Commander, за которыми доступен для работы командный интерпретатор оболочки и выводимая туда информация.
- **Ctrl + x + d** (или через меню **Команда** > **Сравнить каталоги**) позволяет сравнить содержимое каталогов, отображаемых на левой и правой панелях.

Задание 1

1. Откройте Midnight Commander

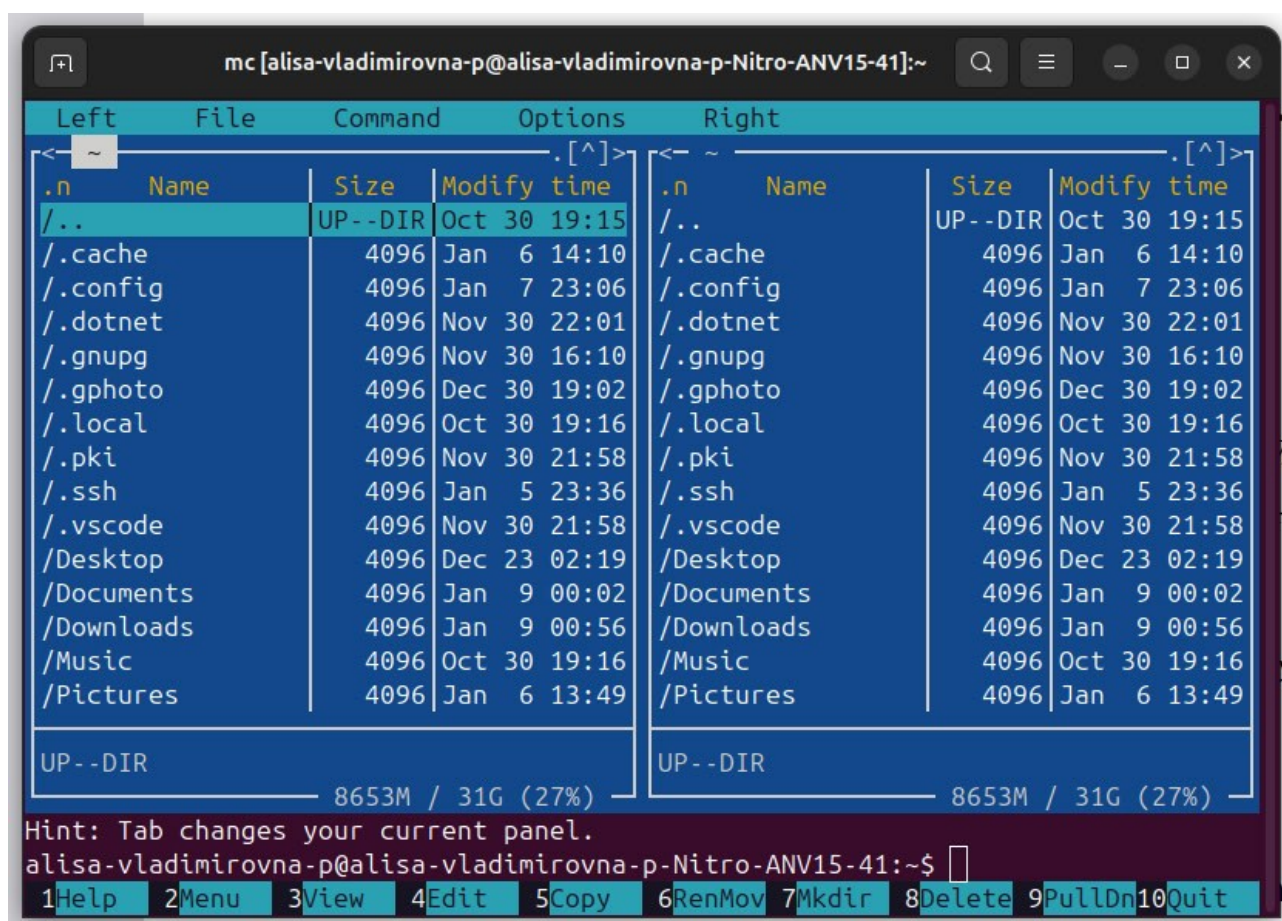
user@dk4n31:~\$ mc

2. Пользуясь клавишами ↑, ↓ и Enter перейдите в каталог ~/work/arch-pc созданный

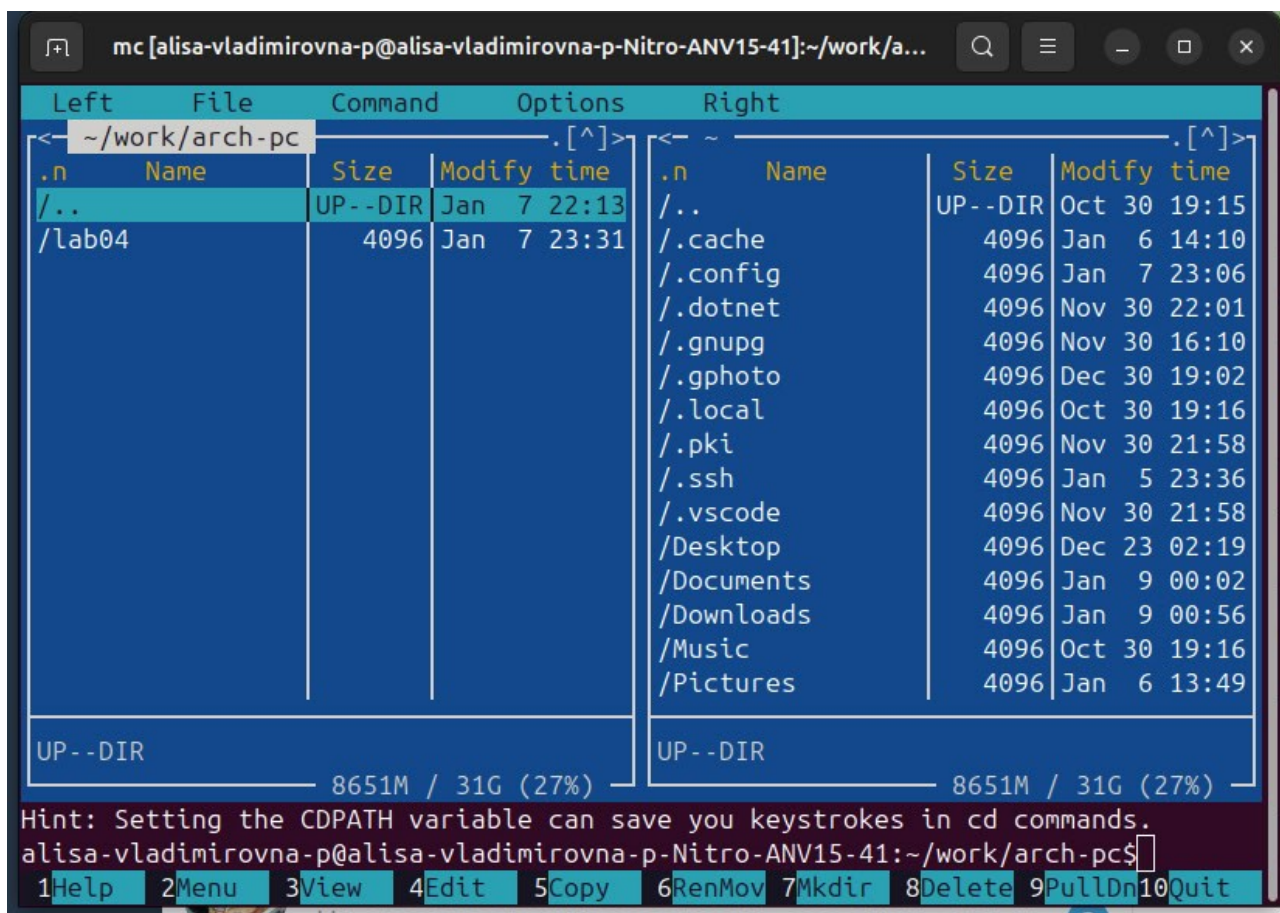
при выполнении лабораторной работы №4 (рис. 5.2).

3. С помощью функциональной клавиши F7 создайте папку lab05 и перейдите в созданный каталог.

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



Открыт Midnight Commander



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

6. Введите текст программы из листинга 5.1 (можно без комментариев), сохраните изменения и закройте файл.. С помощью функциональной клавиши F3 откройте файл lab5-1.asm для просмотра.

Убедитесь, что файл содержит текст программы.

Далее приведён вид выполненного задания где вставлен текст листинга 5.1 без комментариев.

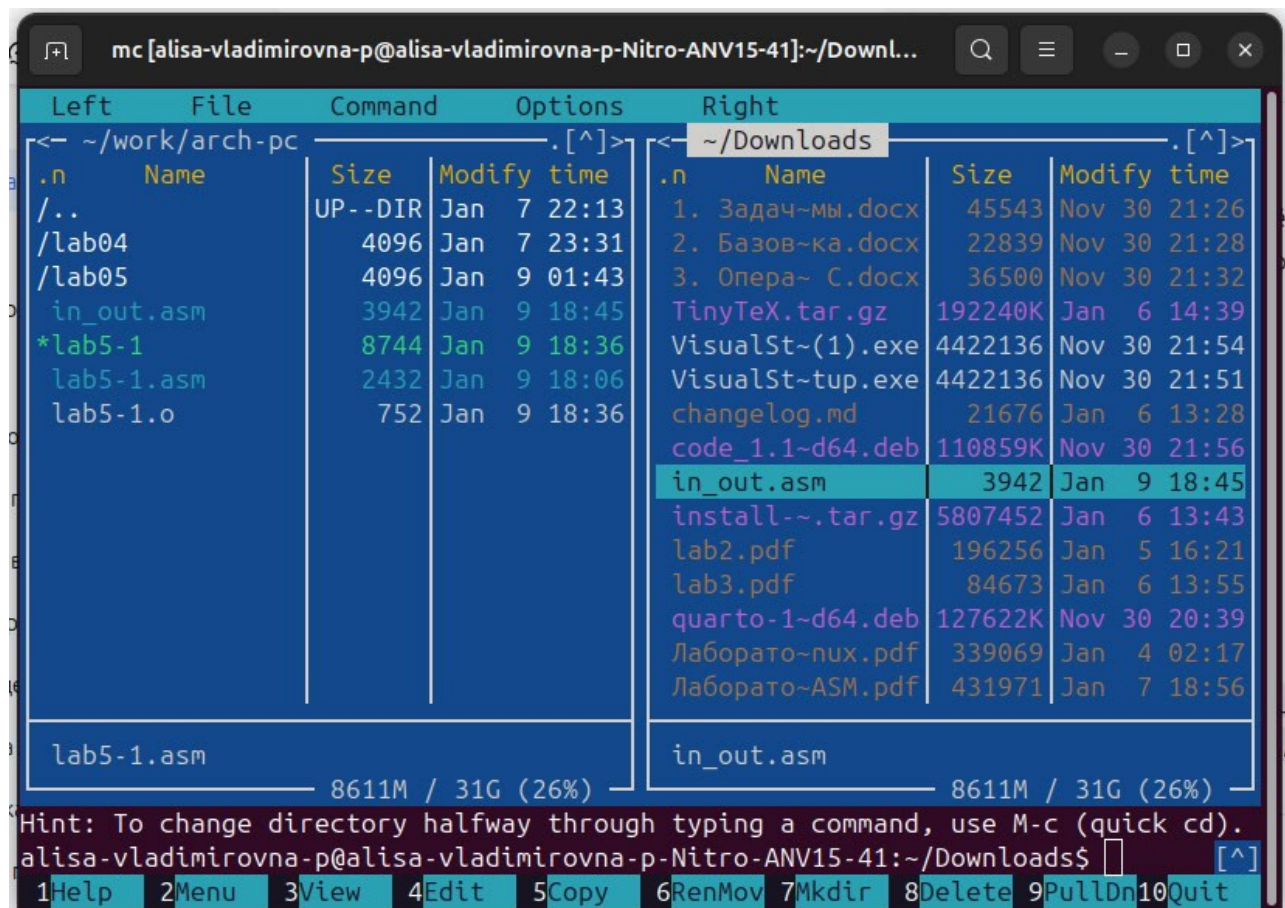
```
/home/alisa-vladimirovna-p/work/arch-pc/lab05/lab5-1.asm *
msgLen equ $-msg
SECTION .bss
buf1 resb 80
SECTION .text
global _start
_start:
    mov eax,4
    mov ebx,1
    mov ecx,msg
    mov edx,msgLen
    int 0x80
    mov eax,3
    mov ebx,0
    mov ecx,buf1
    mov edx,80
    int 0x80
    mov eax,1
    mov ebx,0
    int 0x80

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

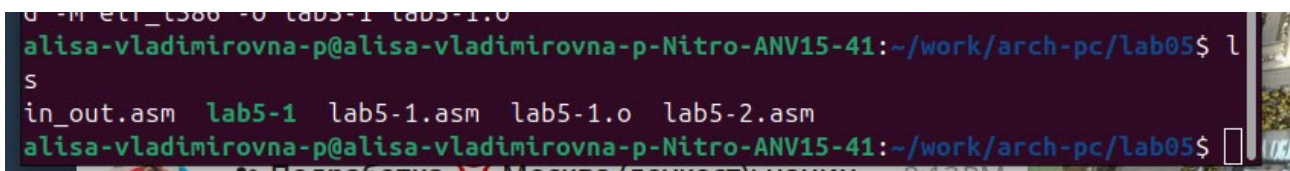
8. Оттранслируйте текст программы lab5-1.asm в объектный файл. Выполните компоновку объектного файла и запустите получившийся исполняемый файл.

```
alisa-vladimirovna-p@alisa-vladimirovna-p-Nitro-ANV15-41:~$ mc
alisa-vladimirovna-p@alisa-vladimirovna-p-Nitro-ANV15-41:~/work/arch-pc$ ./lab5-1
Введите строку:
```


9. Скачайте файл in_out.asm со страницы курса в ТУИС.
10. Подключаемый файл in_out.asm должен лежать в том же каталоге, что и файл с программой, в которой он используется.



11. 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



Приведён список программ в папке на момент выполнения задания.

12. Исправьте текст программы в файле lab5-2.asm с использование 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.
13. В файле lab5-2.asm замените подпрограмму `sprintLF` на `sprint`. Создайте исполняемый файл и проверьте его работу. В чем разница?


```
/home/alisa-vladimirovna-p/work/arch-pc/lab05/lab5-2.asm
%include 'in_out.asm'
SECTION .data
msg: DB 'Введите строку: ',0h
SECTION .bss
buf1: RESB 80
SECTION .text
GLOBAL _start
_start:
    mov eax, msg
    call sprintf
    mov ecx, buf1
    mov edx, 80
    call sread
    call quit
```

[Read 14 lines]

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

```
mc [alisa-vladimirovna-p@alisa-vladimirovna-p-Nitro-ANV15-41]:~/work/a...
alisa-vladimirovna-p@alisa-vladimirovna-p-Nitro-ANV15-41:~$ mc
alisa-vladimirovna-p@alisa-vladimirovna-p-Nitro-ANV15-41:~/work/arch-pc/./lab5-2
Введите строку: 
```

Различие между программами lab5-1 и lab5-2 заключается в положении курсора при вводе строки.

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 введённую строку на экран.

2. Получите исполняемый файл и проверьте его работу. На приглашение 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 введённую строку на экран

```
Введите строку: Алина Владимировна
alisa-vladimirovna-p@alisa-vladimirovna-p-Nitro-ANV15-41:~/work/arch-pc/./lab5-1
-self
Введите строку:
Попова
Попова
```

```
alisa-vladimirovna-p@alisa-vladimirovna-p-Nitro-ANV15-41:~/work/arch-pc/./lab5-2
-self
Введите строку:
ПОПОВА
ПОПОВА
```

```
alisa-vladimirovna-p@alisa-vladimirovna-p-Nitro-ANV15-41: ~/work/study...
alisa-vladimirovna-p@alisa-vladimirovna-p-Nitro-ANV15-41:~/work/study/2025-2026/
Архитектура компьютера/arch-pc$ git push
Username for 'https://github.com': sirokchek
Password for 'https://sirokchek@github.com':
Enumerating objects: 36, done.
Counting objects: 100% (36/36), done.
Delta compression using up to 12 threads
Compressing objects: 100% (30/30), done.
Writing objects: 100% (32/32), 123.83 MiB | 19.54 MiB/s, done.
Total 32 (delta 15), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (15/15), completed with 1 local object.
remote: error: Trace: 647e03662ba529675d9d1f80699332b836a86e87e96e1c09c469966d34
0e7c12
remote: error: See https://gh.io/lfs for more information.
remote: error: File labs/lab03/report/quarto-1.7.34-linux-amd64.deb is 124.63 MB
; this exceeds GitHub's file size limit of 100.00 MB
remote: error: GH001: Large files detected. You may want to try Git Large File S
torage - https://git-lfs.github.com.
To https://github.com/sirokchek/study_2025-2026_arh-pc.git
! [remote rejected] master -> master (pre-receive hook declined)
error: failed to push some refs to 'https://github.com/sirokchek/study_2025-2026
_arh-pc.git'
alisa-vladimirovna-p@alisa-vladimirovna-p-Nitro-ANV15-41:~/work/study/2025-2026/
Архитектура компьютера/arch-pc$
```

Во время попытки задокументировать файлы, произошла ошибка, устранить не удалось.

Выводы

В ходе работы освоены основы работы в Midnight Commander, изучена структура программ на NASM, практиковались системные вызовы `write` и `read`. Приобретены навыки написания, трансляции и отладки программ на ассемблере, работы с вне

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, 2017.
11. Новожилов О. П. Архитектура ЭВМ и систем. — М. : Юрайт, 2016.
12. Расширенный ассемблер: NASM. — 2021. — URL: <https://www.opennet.ru/docs/RUS/nasm/>.
13. Робачевский А., Немнюгин С., Стесик О. Операционная система UNIX. — 2-е изд. — БХВ-Петербург, 2010. — 656 с. — ISBN 978-5-94157-538-1.
14. 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.
15. Таненбаум Э. Архитектура компьютера. — 6-е изд. — СПб. : Питер, 2013. — 874 с. — (Классика Computer Science).
16. Таненбаум Э., Бос Х. Современные операционные системы. — 4-е изд. — СПб. : Питер, 2015. — 1120 с. — (Классика Computer Science).