

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра прикладной информатики и теории вероятностей

ОТЧЕТ

ПО ЛАБОРАТОРНОЙ РАБОТЕ № 10

дисциплина: Архитектура компьютера

Студент: Крыловецкий Денис Витальевич

Группа: НКАбд-03-25

МОСКВА

2025 г.

Оглавление

| | |
|---|----|
| <u>1 Цель работы</u> | 4 |
| <u>2 Задания</u> | 5 |
| <u>3 Теоретическое введение</u> | 6 |
| <u>4 Выполнение работы</u> | 7 |
| <u>Порядок выполнения лабораторной работы</u> | 7 |
| <u>Задание для самостоятельной работы</u> | 12 |
| <u>5 Выводы</u> | 15 |
| <u>Список литературы</u> | 16 |

Список изображений

| | |
|--|----|
| <u>Рисунок 0.1(создание каталога и переход в него)</u> | 7 |
| <u>Рисунок 0.2(создание файлов lab10-1.asm, readme-1.txt и readme-2.txt)</u> | 7 |
| <u>Рисунок 0.3(код программы)</u> | 8 |
| <u>Рисунок 0.4(код программы)</u> | 8 |
| <u>Рисунок 0.5(создание исполняемого файла)</u> | 9 |
| <u>Рисунок 0.6(создание исполняемого файла)</u> | 9 |
| <u>Рисунок 0.7(запуск исполняемого файла)</u> | 9 |
| <u>Рисунок 0.8(проверка работы файла)</u> | 9 |
| <u>Рисунок 0.9(проверка работы файла)</u> | 9 |
| <u>Рисунок 0.10(команда chmod, запрещение выполнения)</u> | 9 |
| <u>Рисунок 0.11(выполнение файла)</u> | 10 |
| <u>Рисунок 0.12 (изменение прав доступа с помощью команды chmod)</u> | 10 |
| <u>Рисунок 0.13(выполнение кода)</u> | 11 |
| <u>Рисунок 0.14(предоставление прав в символьном виде)</u> | 11 |
| <u>Рисунок 0.15(проверка)</u> | 11 |
| <u>Рисунок 0.16(предоставление прав в двоичном виде)</u> | 12 |
| <u>Рисунок 0.17(проверка)</u> | 12 |
| <u>Рисунок 0.18(создание файла)</u> | 12 |
| <u>Рисунок 0.19(Код программы)</u> | 13 |
| <u>Рисунок 0.20(Код программы)</u> | 14 |
| <u>Рисунок 0.21(создание исполняемого файла)</u> | 14 |
| <u>Рисунок 0.22(создание исполняемого файла)</u> | 14 |
| <u>Рисунок 0.23(Проверка работы)</u> | 14 |
| <u>Рисунок 0.24(проверка наличия файла)</u> | 14 |
| <u>Рисунок 0.25(проверка содержимого файла)</u> | 14 |

1 Цель работы

Приобретение навыков написания программ для работы с файлами.

2 Задания

Создание файлов в программах, изменение прав на работу с файлами для различных пользователей, задания для самостоятельной работы.

3 Теоретическое введение

Операционная система GNU/Linux является многопользовательской.

Для защиты данных каждого пользователя от несанкционированного доступа со стороны других пользователей в системе предусмотрены механизмы **разграничения прав доступа к файлам и каталогам.**

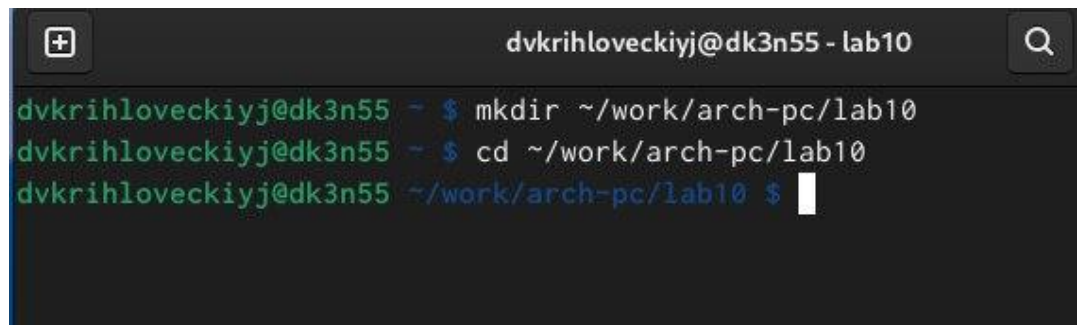
Эти механизмы не только ограничивают доступ, но и позволяют **настраивать разрешения** для совместной работы с данными — например, предоставлять доступ к файлам отдельным пользователям или группам пользователей.

Таким образом, система безопасности Linux обеспечивает как **конфиденциальность** данных пользователей, так и **гибкие возможности** для коллективного использования ресурсов.

4 Выполнение работы

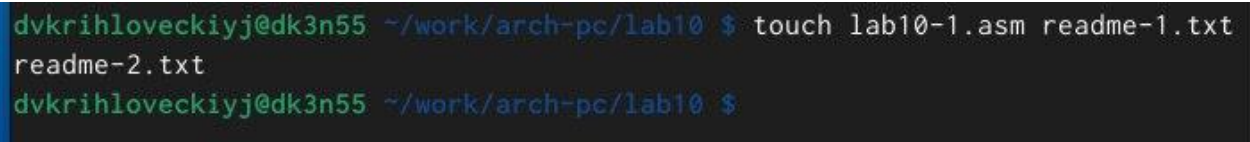
Ход выполнения лабораторной работы

1. Создадим каталог для программ лабораторной работы № 10, перейдем в него и создадим файлы lab10-1.asm, readme-1.txt и readme-2.txt (рисунок 0.1-0.2)



```
dvkrihloveckiyj@dk3n55 - lab10
dvkrihloveckiyj@dk3n55 ~ $ mkdir ~/work/arch-pc/lab10
dvkrihloveckiyj@dk3n55 ~ $ cd ~/work/arch-pc/lab10
dvkrihloveckiyj@dk3n55 ~/work/arch-pc/lab10 $
```

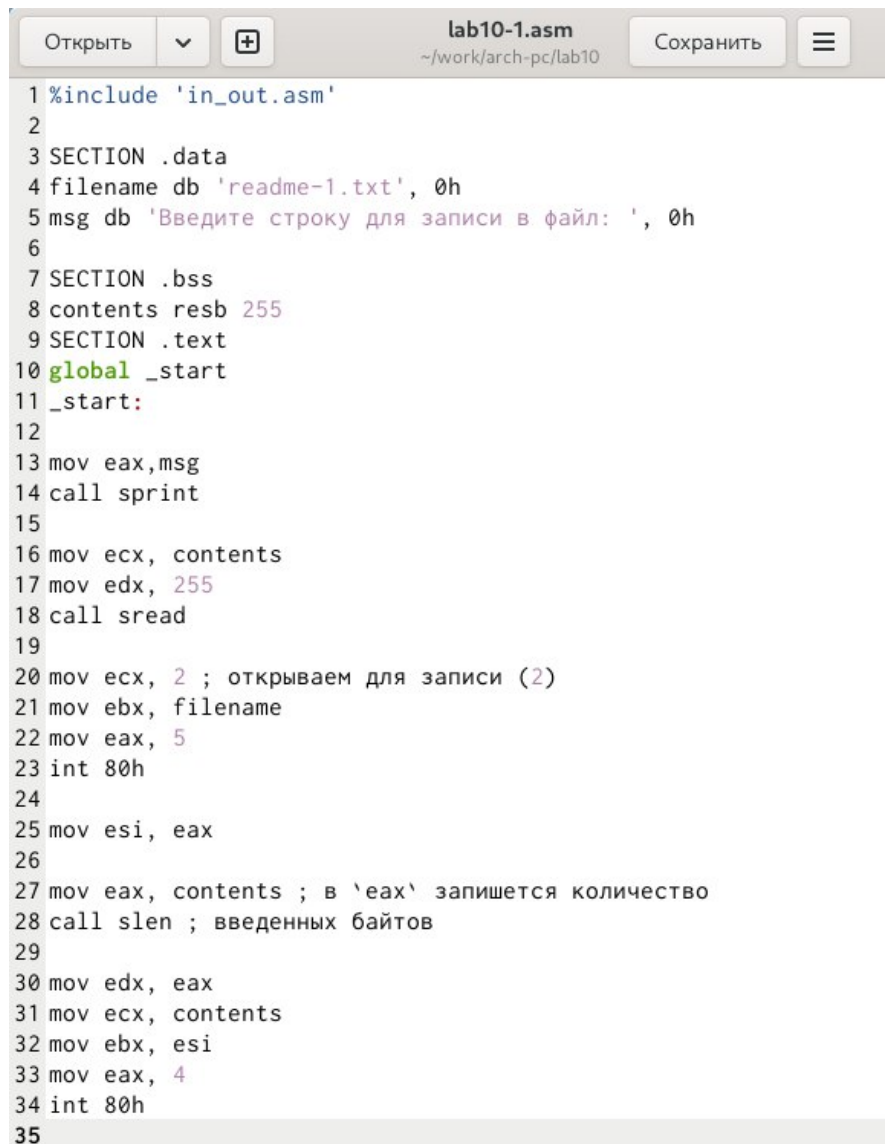
Рисунок 0.1(создание каталога и переход в него)



```
dvkrihloveckiyj@dk3n55 ~/work/arch-pc/lab10 $ touch lab10-1.asm readme-1.txt
readme-2.txt
dvkrihloveckiyj@dk3n55 ~/work/arch-pc/lab10 $
```

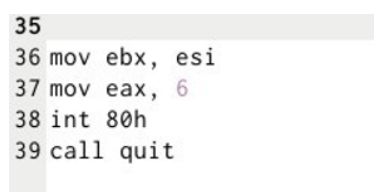
Рисунок 0.2(создание файлов lab10-1.asm, readme-1.txt и readme-2.txt)

2. Введем в файл lab10-1.asm текст программы из листинга (рисунок 0.3-0.4)



```
1 %include 'in_out.asm'
2
3 SECTION .data
4 filename db 'readme-1.txt', 0h
5 msg db 'Введите строку для записи в файл: ', 0h
6
7 SECTION .bss
8 contents resb 255
9 SECTION .text
10 global _start
11 _start:
12
13 mov eax,msg
14 call sprint
15
16 mov ecx, contents
17 mov edx, 255
18 call sread
19
20 mov ecx, 2 ; открываем для записи (2)
21 mov ebx, filename
22 mov eax, 5
23 int 80h
24
25 mov esi, eax
26
27 mov eax, contents ; в 'eax' запишется количество
28 call slen ; введенных байтов
29
30 mov edx, eax
31 mov ecx, contents
32 mov ebx, esi
33 mov eax, 4
34 int 80h
35
```

Рисунок 0.3(код программы)



```
35
36 mov ebx, esi
37 mov eax, 6
38 int 80h
39 call quit
```

Рисунок 0.4(код программы)

Создадим исполняемый файл и проверим его работу(рисунки 0.5-0.9)

```
dvkrihloveckiyj@dk3n55 ~/work/arch-pc/lab10 $ nasm -f elf -g -l lab10-1.lst lab10-1.asm
```

Рисунок 0.5(создание исполняемого файла)

```
dvkrihloveckiyj@dk3n55 ~/work/arch-pc/lab10 $ ld -m elf_i386 -o lab10-1 lab10-1.o
dvkrihloveckiyj@dk3n55 ~/work/arch-pc/lab10 $
```

Рисунок 0.6(создание исполняемого файла)

```
dvkrihloveckiyj@dk3n55 ~/work/arch-pc/lab10 $ ./lab10-1
Введите строку для записи в файл: krihloveckiyj
```

Рисунок 0.7(запуск исполняемого файла)

```
dvkrihloveckiyj@dk3n55 ~/work/arch-pc/lab10 $ ./lab10-1
Введите строку для записи в файл: krihloveckiyj
dvkrihloveckiyj@dk3n55 ~/work/arch-pc/lab10 $ ls -l
итого 32
-rw-r--r-- 1 dvkrihloveckiyj studsci 3942 дек 17 19:52 in_out.asm
-rwxr-xr-x 1 dvkrihloveckiyj studsci 9772 дек 25 14:00 lab10-1
-rw-r--r-- 1 dvkrihloveckiyj studsci 610 дек 25 14:00 lab10-1.asm
-rw-r--r-- 1 dvkrihloveckiyj studsci 12996 дек 25 14:00 lab10-1.lst
-rw-r--r-- 1 dvkrihloveckiyj studsci 2560 дек 25 14:00 lab10-1.o
-rw-r--r-- 1 dvkrihloveckiyj studsci 14 дек 25 14:01 readme-1.txt
-rw-r--r-- 1 dvkrihloveckiyj studsci 0 дек 25 13:53 readme-2.txt
dvkrihloveckiyj@dk3n55 ~/work/arch-pc/lab10 $
```

Рисунок 0.8(проверка работы файла)

```
dvkrihloveckiyj@dk3n55 ~/work/arch-pc/lab10 $ cat readme-1.txt
krihloveckiyj
dvkrihloveckiyj@dk3n55 ~/work/arch-pc/lab10 $
```

Рисунок 0.9(проверка работы файла)

3. С помощью команды `chmod` изменим права доступа к исполняемому файлу `lab10-1`, запретив его выполнение. (рисунок 0.10)

```

dvkrihloveckiyj@dk3n55 ~/work/arch-pc/lab10 $ ls -l
итого 32
-rw-r--r-- 1 dvkrihloveckiyj studsci 3942 дек 17 19:52 in_out.asm
-rwxr-xr-x 1 dvkrihloveckiyj studsci 9772 дек 25 14:00 lab10-1
-rw-r--r-- 1 dvkrihloveckiyj studsci 610 дек 25 14:00 lab10-1.asm
-rw-r--r-- 1 dvkrihloveckiyj studsci 12996 дек 25 14:00 lab10-1.lst
-rw-r--r-- 1 dvkrihloveckiyj studsci 2560 дек 25 14:00 lab10-1.o
-rw-r--r-- 1 dvkrihloveckiyj studsci 14 дек 25 14:01 readme-1.txt
-rw-r--r-- 1 dvkrihloveckiyj studsci 0 дек 25 13:53 readme-2.txt
dvkrihloveckiyj@dk3n55 ~/work/arch-pc/lab10 $ chmod a-x lab10-1
dvkrihloveckiyj@dk3n55 ~/work/arch-pc/lab10 $

```

Рисунок 0.10(команда chmod, запрещение выполнения)

Попробуем выполнить файл. (рисунок 0.11)

```

dvkrihloveckiyj@dk3n55 ~/work/arch-pc/lab10 $ chmod a-x lab10-1
dvkrihloveckiyj@dk3n55 ~/work/arch-pc/lab10 $ ./lab10-1
bash: ./lab10-1: Отказано в доступе
dvkrihloveckiyj@dk3n55 ~/work/arch-pc/lab10 $

```

Рисунок 0.11(выполнение файла)

Система не позволяет запустить файл, потому что у него отсутствует бит выполнения (x). Даже если файл содержит корректный исполняемый код, ОС проверяет права доступа перед запуском. Без флага x файл рассматривается только как данные для чтения/записи.

4.С помощью команды chmod изменим права доступа к файлу lab10-1.asm с исходным текстом программы, добавив права на исполнение.(рисунок 0.12)

```

dvkrihloveckiyj@dk3n55 ~/work/arch-pc/lab10 $ ls -l
итого 32
-rw-r--r-- 1 dvkrihloveckiyj studsci 3942 дек 17 19:52 in_out.asm
-rw-r--r-- 1 dvkrihloveckiyj studsci 9772 дек 25 14:00 lab10-1
-rw-r--r-- 1 dvkrihloveckiyj studsci 610 дек 25 14:00 lab10-1.asm
-rw-r--r-- 1 dvkrihloveckiyj studsci 12996 дек 25 14:00 lab10-1.lst
-rw-r--r-- 1 dvkrihloveckiyj studsci 2560 дек 25 14:00 lab10-1.o
-rw-r--r-- 1 dvkrihloveckiyj studsci 14 дек 25 14:01 readme-1.txt
-rw-r--r-- 1 dvkrihloveckiyj studsci 0 дек 25 13:53 readme-2.txt
dvkrihloveckiyj@dk3n55 ~/work/arch-pc/lab10 $ chmod a+x lab10-1.asm
dvkrihloveckiyj@dk3n55 ~/work/arch-pc/lab10 $

```

Рисунок 0.12 (изменение прав доступа с помощью команды chmod)

Попытаемся выполнить его (рисунок 0.13)

```
dvkrihloveckiyj@dk3n55 ~/work/arch-pc/lab10 $ ./lab10-1.asm
./lab10-1.asm: строка 1: fg: нет управления заданиями
./lab10-1.asm: строка 3: SECTION: команда не найдена
./lab10-1.asm: строка 4: filename: команда не найдена
./lab10-1.asm: строка 5: msg: команда не найдена
./lab10-1.asm: строка 7: SECTION: команда не найдена
./lab10-1.asm: строка 8: contents: команда не найдена
./lab10-1.asm: строка 9: SECTION: команда не найдена
./lab10-1.asm: строка 10: global: команда не найдена
./lab10-1.asm: строка 11: _start:: команда не найдена
./lab10-1.asm: строка 13: mov: команда не найдена
./lab10-1.asm: строка 14: call: команда не найдена
./lab10-1.asm: строка 16: mov: команда не найдена
./lab10-1.asm: строка 17: mov: команда не найдена
./lab10-1.asm: строка 18: call: команда не найдена
./lab10-1.asm: строка 20: синтаксическая ошибка рядом с неожиданным маркером «(»
./lab10-1.asm: строка 20: `mov ecx, 2 ; открываем для записи (2) '
dvkrihloveckiyj@dk3n55 ~/work/arch-pc/lab10 $
```

Рисунок 0.13(выполнение кода)

Поскольку ни одна из строк не представляет собой команду Bash, программа фактически ничего не выполняет.

5.В соответствии с вариантом 2 предоставим права доступа к файлу readme 1.txt представленные в символьном виде, а для файла readme-2.txt–в двоичном виде.(рисунок 0.14-0.17)

```
dvkrihloveckiyj@dk3n55 ~/work/arch-pc/lab10 $ chmod u=rx,g=rwx,o=rwx readme-1.txt
dvkrihloveckiyj@dk3n55 ~/work/arch-pc/lab10 $
```

Рисунок 0.14(предоставление прав в символьном виде)

```
dvkrihloveckiyj@dk3n55 ~/work/arch-pc/lab10 $ ls -l readme-1.txt
-r-xrwxrwx 1 dvkrihloveckiyj studsci 14 дек 25 14:01 readme-1.txt
dvkrihloveckiyj@dk3n55 ~/work/arch-pc/lab10 $
```

Рисунок 0.15(проверка)

```
dvkrihloveckiyj@dk3n55 ~/work/arch-pc/lab10 $ chmod 577 readme-2.txt
dvkrihloveckiyj@dk3n55 ~/work/arch-pc/lab10 $
```

Рисунок 0.16(предоставление прав в двоичном виде)

```
dvkrihloveckiyj@dk3n55 ~/work/arch-pc/lab10 $ ls -l readme-2.txt
-r-xrwxrwx 1 dvkrihloveckiyj studsci 0 дек 25 13:53 readme-2.txt
dvkrihloveckiyj@dk3n55 ~/work/arch-pc/lab10 $
```

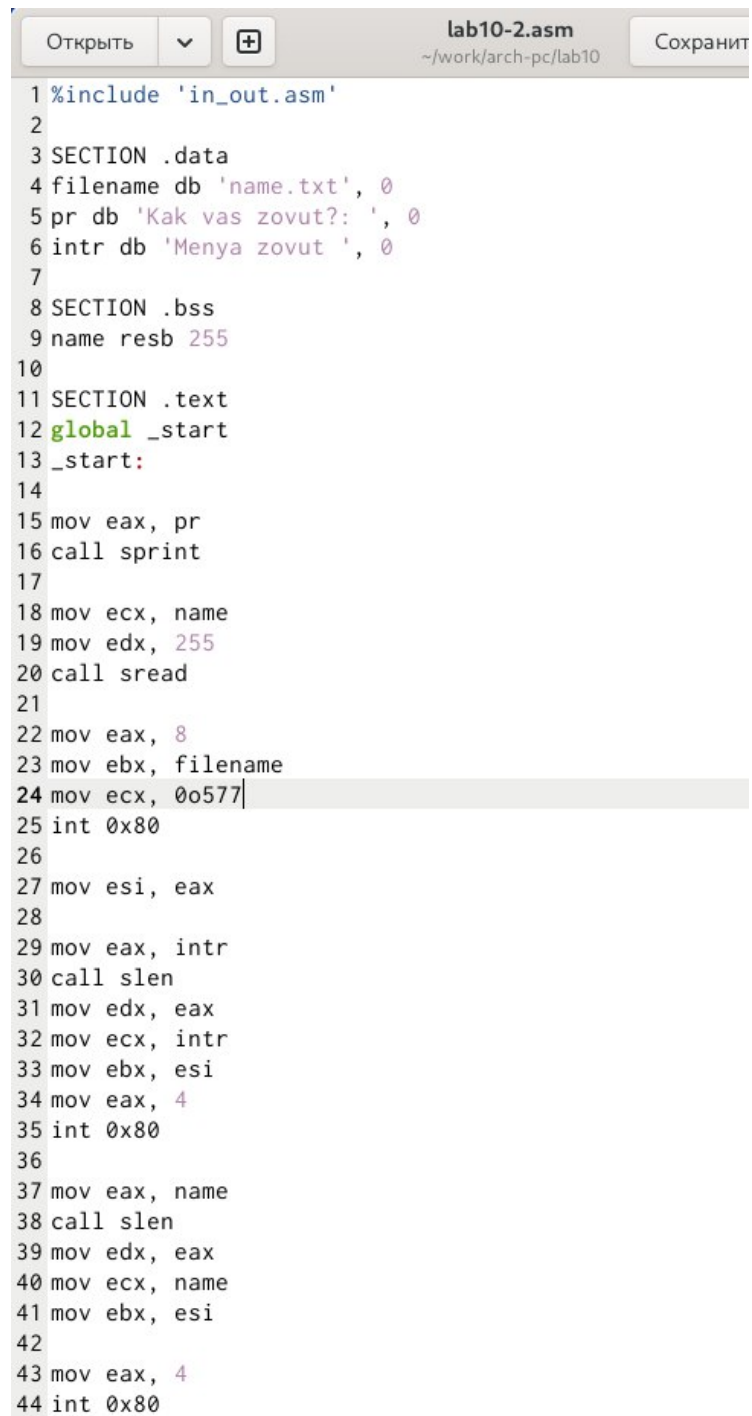
Рисунок 0.17(проверка)

Задание для самостоятельной работы

1.Напишем программу (рисунок 0.18-0.20)

```
dvkrihloveckiyj@dk3n55 ~/work/arch-pc/lab10 $ touch lab10-2.asm
dvkrihloveckiyj@dk3n55 ~/work/arch-pc/lab10 $
```

Рисунок 0.18(создание файла)



```
1 %include 'in_out.asm'
2
3 SECTION .data
4 filename db 'name.txt', 0
5 pr db 'Kak vas zovut?: ', 0
6 intr db 'Menya zovut ', 0
7
8 SECTION .bss
9 name resb 255
10
11 SECTION .text
12 global _start
13 _start:
14
15 mov eax, pr
16 call sprint
17
18 mov ecx, name
19 mov edx, 255
20 call sread
21
22 mov eax, 8
23 mov ebx, filename
24 mov ecx, 0x577
25 int 0x80
26
27 mov esi, eax
28
29 mov eax, intr
30 call slen
31 mov edx, eax
32 mov ecx, intr
33 mov ebx, esi
34 mov eax, 4
35 int 0x80
36
37 mov eax, name
38 call slen
39 mov edx, eax
40 mov ecx, name
41 mov ebx, esi
42
43 mov eax, 4
44 int 0x80
```

Рисунок 0.19(Код программы)

```

45
46 mov ebx, esi
47 mov eax, 8
48 int 0x80
49
50 call quit

```

Рисунок 0.20(Код программы)

Создадим исполняемый файл и проверим его работу (рисунок 0.21-0.23)

```

dvkrihloveckiyj@dk3n55 ~/work/arch-pc/lab10 $ nasm -f elf -g -l lab10-2.lst lab10-2.asm

```

Рисунок 0.21(создание исполняемого файла)

```

dvkrihloveckiyj@dk3n55 ~/work/arch-pc/lab10 $ nasm -f elf -g -l lab10-2.lst lab10-2.asm
dvkrihloveckiyj@dk3n55 ~/work/arch-pc/lab10 $ ld -m elf_i386 -o lab10-2 lab10-2.o
dvkrihloveckiyj@dk3n55 ~/work/arch-pc/lab10 $

```

Рисунок 0.22(создание исполняемого файла)

```

dvkrihloveckiyj@dk3n55 ~/work/arch-pc/lab10 $ ./lab10-2
Kak vas zovut?: Denis
dvkrihloveckiyj@dk3n55 ~/work/arch-pc/lab10 $

```

Рисунок 0.23(Проверка работы)

Проверим наличие файла и его содержимое с помощью команд ls и cat(рисунок 0.24-0.25)

```

dvkrihloveckiyj@dk3n55 ~/work/arch-pc/lab10 $ ls
in_out.asm  lab10-1.asm  lab10-1.o  lab10-2.asm  lab10-2.o  readme-1.txt
lab10-1     lab10-1.lst  lab10-2   lab10-2.lst  name.txt   readme-2.txt
dvkrihloveckiyj@dk3n55 ~/work/arch-pc/lab10 $

```

Рисунок 0.24(проверка наличия файла)

```

dvkrihloveckiyj@dk3n55 ~/work/arch-pc/lab10 $ cat name.txt
Menya zovut Denis
dvkrihloveckiyj@dk3n55 ~/work/arch-pc/lab10 $

```

Рисунок 0.25(проверка содержимого файла)

5 Выводы

Я приобрел навыки написания программ для работы с файлами.

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](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 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).