

Отчёт по лабораторной работе №5

Архитектура компьютера

Агапова Анна Антоновна

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
3	Выводы	11
4	Список литературы	12

Список иллюстраций

2.1	Открытие Midnight Commander	6
2.2	Переход в каталог	6
2.3	Создание папки.....	7
2.4	Переход в каталог	7
2.5	Создание файла.....	7
2.6	Проверка создания файла.....	7
2.7	Ввод текста программы.....	7
2.8	Проверка наличия текста программы.....	8
2.9	Транслирование текста, проверка работы	8
2.10	Копирование файла	8
2.11	Проверка копирования файла	9
2.12	Создание копии файла.....	9
2.13	Исправление текста программы.....	9
2.14	Проверка работы.....	9
2.15	Проверка работы.....	10
2.16	Проверка работы.....	10

Список таблиц

1 Цель работы

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

2 Выполнение лабораторной работы

1. Открываю Midnight Commander с помощью команды mc. (рис. 2.1).

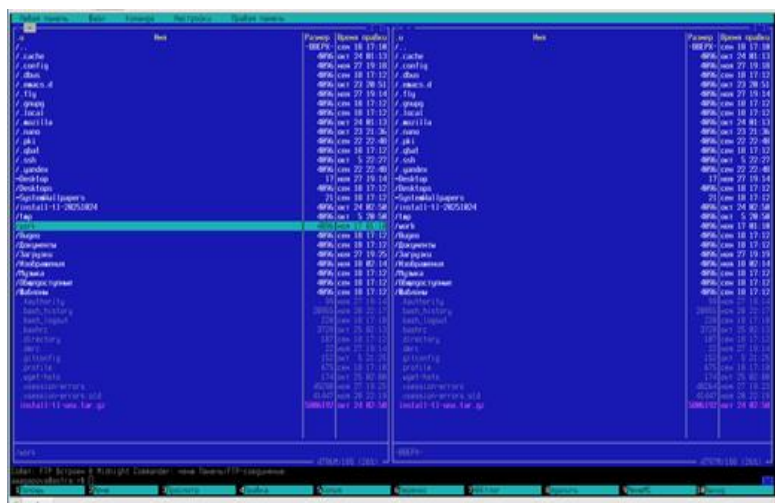


Рисунок 2.1: Открытие Midnight Commander

2. Пользуясь клавишами клавиатуры перехожу в каталог ~/work/arch-rc созданный при выполнении лабораторной работы №4. (рис. 2.2).

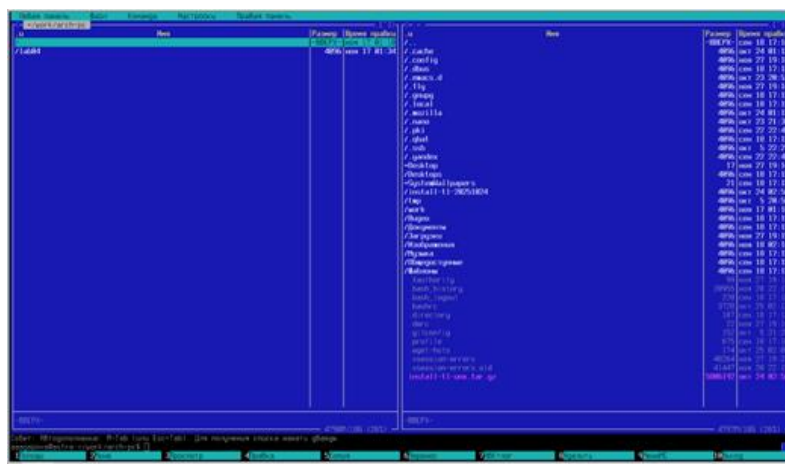


Рисунок 2.2: Переход в каталог

3. С помощью функциональной клавиши F7 создаю папку lab05 (рис. 2.3) и перехожу в созданный каталог. (рис. 2.4)

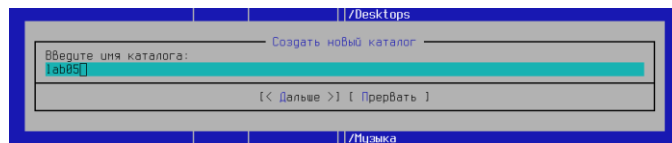


Рисунок 2.3: Создание папки

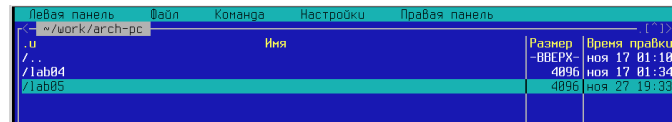


Рисунок 2.4: Переход в каталог

4. Пользуясь строкой ввода и командой touch создаю файл lab5-1.asm. (рис. 2.5)

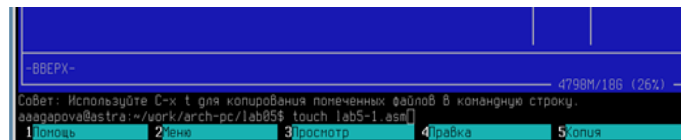


Рисунок 2.5: Создание файла

5. Проверю, что файл lab5-1.asm создан. (рис. 2.6).

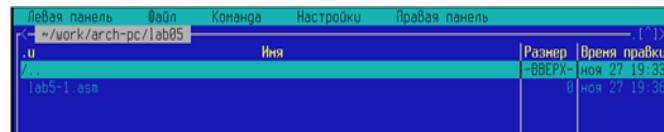


Рисунок 2.6: Проверка создания файла

6. С помощью функциональной клавиши F4 открываю файл lab5-1.asm для редактирования во встроенном редакторе. Ввожу текст программы из листинга 5.1, сохраняю изменения и закрываю файл (рис. 2.7)

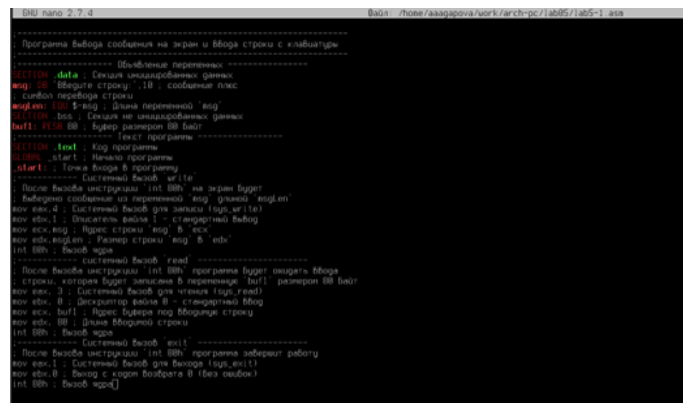


Рисунок 2.7: Ввод текста программы

7. С помощью функциональной клавиши F3 открываю файл lab5-1.asm для просмотра. Убеждаюсь, что файл содержит текст программы. (рис. 2.8).

```

/home/aaagapova/work/arch-pc/lab05/lab5-1.asm
; 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
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 ; Файловый дескриптор 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 ; Дескриптор файла 0 - стандартный ввод
mov ecx,buf1 ; Адрес буфера по которому строку
mov edx,80 ; Длина 80-байтовой 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)
int 80h ; Вызов ядра

```

Рисунок 2.8: Проверка наличия текста программы

8. Оттранслирую текст программы lab5-1.asm в объектный файл. Выполняю компоновку объектного файла и запускаю получившийся исполняемый файл (рис. 2.9).

```

aaagapova@astra:~/work/arch-pc/lab05$ nasm -f elf lab5-1.asm
aaagapova@astra:~/work/arch-pc/lab05$ ld -m elf_i386 -o lab5-1 lab5-1.o
aaagapova@astra:~/work/arch-pc/lab05$ ./lab5-1
Введите строку:
Янна Яганава

```

Рисунок 2.9: Транслирование текста, проверка работы

9. Скачиваю файл in_out.asm со страницы курса в ТУИС. В одной из панелей mc открываю каталог с файлом lab5-1.asm. В другой панели каталог со скаченным файлом in_out.asm. Скопирую файл in_out.asm в каталог с файлом lab5-1.asm с помощью функциональной клавиши F5. (рис. 2.10).

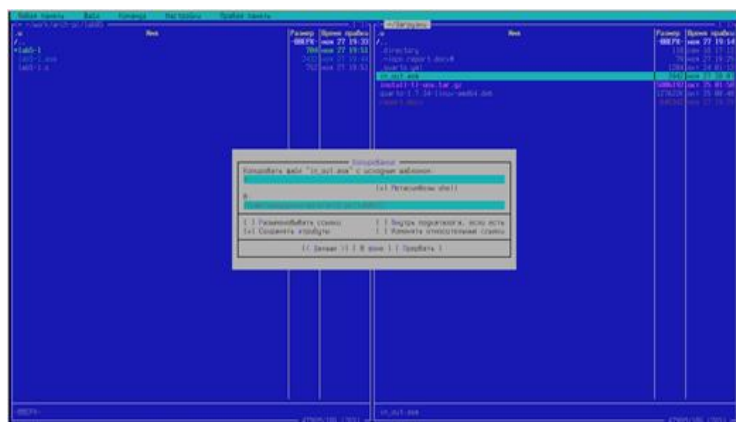
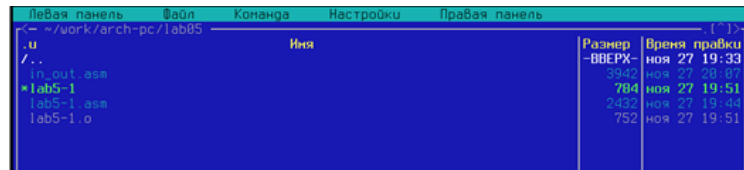


Рисунок 2.10: Копирование файла

10. Проверю, что файл `in_out.asm` скопировался в каталог с файлом `lab5-1.asm`. (рис. 2.11).



Файл	Размер	Время правки
in_out.asm	3942	ноя 27 19:33
lab5-1.asm	784	ноя 27 19:51
lab5-1.o	2432	ноя 27 19:51

Рисунок 2.11: Проверка копирования файла

11. С помощью функциональной клавиши F6 создаю копию файла `lab5-1.asm` с именем `lab5-2.asm`. (рис. 2.12).

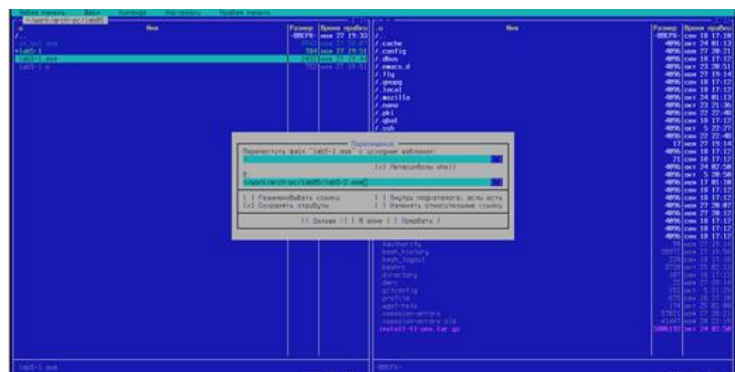
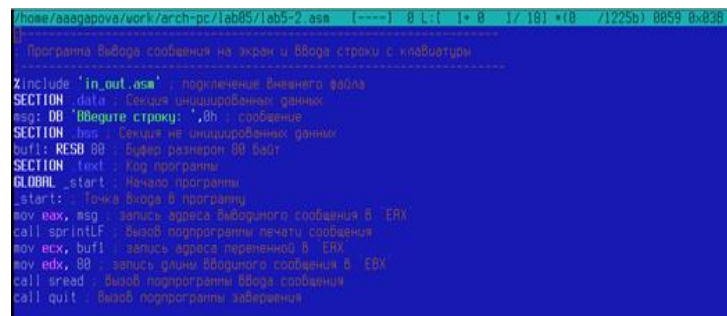


Рисунок 2.12: Создание копии файла

12. Исправляю текст программы в файле `lab5-2.asm` с использованием подпрограмм из внешнего файла `in_out.asm`, используя подпрограммы `sprintLF`, `sread` и `quit` в соответствии с листингом 5.2. (рис. 2.13).



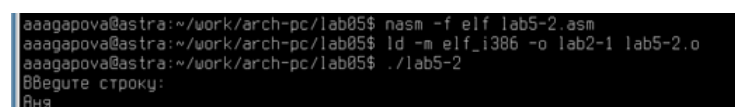
```

/home/aaagapova/work/arch-pc/lab05/lab5-2.asm [---] 8 L: 1* 8 1/ 10) *(8 /12256) 8859 0x836
; Программа Вывода сообщения на экран и Ввода строки с клавиатуры
Zinclude "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, esp ; запись адреса входного сообщения в EAX
call sprintLF ; Вывод подпрограммы печати сообщения
mov ecx, buf1 ; запись адреса переменных в EAX
mov edx, 80 ; запись длины входного сообщения в EDI
call sread ; Вывод подпрограммы ввода сообщения
call quit ; Вывод подпрограммы завершения

```

Рисунок 2.13: Исправление текста программы

13. Проверим работу исправленного текста программы. (рис. 2.14).



```

aaagapova@astra:~/work/arch-pc/lab05$ nasm -f elf lab5-2.asm
aaagapova@astra:~/work/arch-pc/lab05$ ld -m elf_i386 -o lab2-1 lab5-2.o
aaagapova@astra:~/work/arch-pc/lab05$ ./lab2-1
Введите строку:
Яня

```

Рисунок 2.14: Проверка работы

14. Создаю копию файла lab5-1.asm. Внесу изменения в программу (без использования внешнего файла in_out.asm) и проверяю его работу. (рис. 2.15).

```
aaagapova@astra:~/work/arch-pc/lab05$ nasm -f elf lab5-11.asm
aaagapova@astra:~/work/arch-pc/lab05$ ld -m elf_i386 -o lab5-11 lab5-11.o
aaagapova@astra:~/work/arch-pc/lab05$ ./lab5-11
Введите строку:
Агарова
Вы ввели:
Агарова
```

Рисунок 2.15: Проверка работы

15. Создаю копию файла lab5-2.asm. Исправляю текст программы с использование подпрограмм из внешнего файла in_out.asm и проверяю его работу. (рис. 2.16)

```
aaagapova@astra:~/work/arch-pc/lab05$ nasm -f elf lab5-22.asm
aaagapova@astra:~/work/arch-pc/lab05$ ld -m elf_i386 -o lab5-22 lab5-22.o
aaagapova@astra:~/work/arch-pc/lab05$ ./lab5-22
Введите строку:
Агарова
Агарова
```

Рисунок 2.16: Проверка работы

3 Выводы

В ходе лабораторной работы мною были приобретены практические навыки работы в Midnight Commander и освоение инструкций языка ассемблера `mov` и `int`.

4 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/Learningbash-Shell-Programming-Nutshell/dp/0596009658>.
6. Robbins A. Bash Pocket Reference. — O'Reilly Media, 2016. — 156 с. — ISBN 9781491941591.
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. — БХВПетербург, 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).