

Отчёт по лабораторной работе

Простейший вариант

Тимур Ринатович Каримов

Содержание

| | | |
|----------|---------------------------------------|-----------|
| 1 | Цель работы | 5 |
| 2 | Задание | 6 |
| 3 | Теоретическое введение | 7 |
| 4 | Выполнение лабораторной работы | 8 |
| 5 | Выводы | 16 |
| | Список литературы | 17 |

Список иллюстраций

| | | |
|------|--|----|
| 4.1 | Создание папки lab05 | 8 |
| 4.2 | Создание папки lab05 | 9 |
| 4.3 | Файл lab05 в окне редактора mcedit | 9 |
| 4.4 | Файл lab05 в окне редактора mcedit | 10 |
| 4.5 | Выполнение команды F3 в mcedit | 10 |
| 4.6 | Выполнение команды F3 в mcedit | 11 |
| 4.7 | Указание пути для копирования файла в mcedit | 11 |
| 4.8 | Исправление текст программы lab5-2.asm в соответствии с листин- гом 5.2 | 11 |
| 4.9 | Запуск файла lab5-2.asm для проверки | 12 |
| 4.10 | Редактирование подпрограммы sprintLF. Замена на sprint | 12 |
| 4.11 | Запуск полученного файла | 12 |
| 4.12 | Указание пути для копирования файла в mcedit | 13 |
| 4.13 | Внесение изменения в соответствии с алгоритмом | 13 |
| 4.14 | Запуск файла lab5-11.asm | 14 |
| 4.15 | Указание пути для копирования файла в mcedit | 14 |
| 4.16 | Внесение изменения в соответствии с алгоритмом | 15 |
| 4.17 | Запуск файла lab5-11.asm | 15 |

Список таблиц

| | | |
|-----|---|---|
| 3.1 | Описание некоторых каталогов файловой системы GNU Linux . . . | 7 |
|-----|---|---|

1 Цель работы

Цель данной лабораторной работы заключается в изучении основ работы с текстовым интерфейсом файлового менеджера Midnight Commander (mc), а также в освоении структуры и синтаксиса программы на языке ассемблера NASM. В ходе выполнения лабораторной работы студенты познакомятся с различными командами и операциями, доступными в mc, а также научатся работать с данными в секциях bss и data, а также разберутся в использовании инструкций языка ассемблера.

2 Задание

- 1) Выполнение лабораторной работы
- 2) Подключение внешнего файла in_out.asm
- 3) Выполнения самостоятельной работы

3 Теоретическое введение

Здесь описываются теоретические аспекты, связанные с выполнением работы.

Например, в табл. 3.1 приведено краткое описание стандартных каталогов Unix.

Таблица 3.1: Описание некоторых каталогов файловой системы GNU Linux

| Имя каталога | Описание каталога |
|--------------|--|
| / | Корневая директория, содержащая всю файловую систему |
| /bin | Основные системные утилиты, необходимые как в однопользовательском режиме, так и при обычной работе всем пользователям |
| /etc | Общесистемные конфигурационные файлы и файлы конфигурации установленных программ |
| /home | Содержит домашние директории пользователей, которые, в свою очередь, содержат персональные настройки и данные пользователя |
| /media | Точки монтирования для сменных носителей |
| /root | Домашняя директория пользователя root |
| /tmp | Временные файлы |
| /usr | Вторичная иерархия для данных пользователя |

Более подробно про Unix см. в [1–4].

4 Выполнение лабораторной работы

Откроем Midnight commander с помощью команды *mc*, перейдем в каталог `~/work/arch-pc` и создадим папку `lab05` (рис. 4.1).

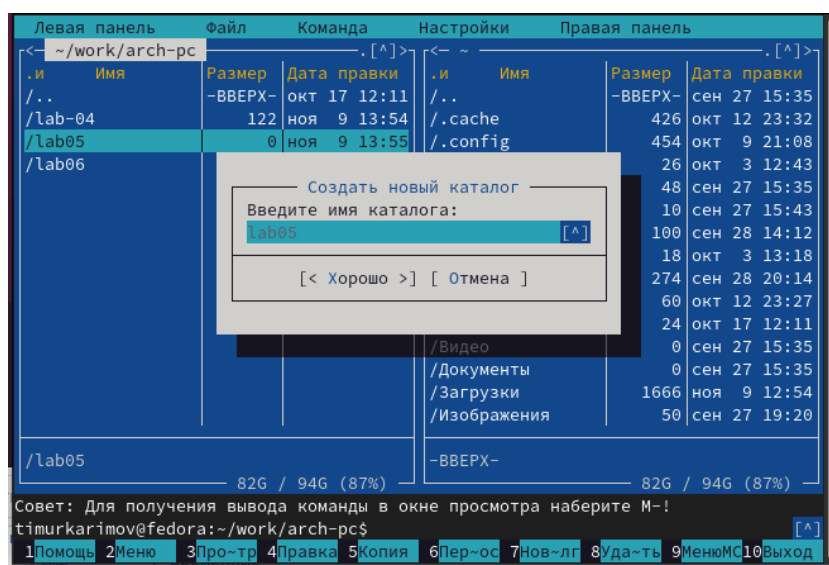


Рис. 4.1: Создание папки `lab05`

Создадим файл `lab5-1.ams` с помощью команды *touch* (рис. 4.2).


```

lab5-1.asm      [----] 32 L: [ 1+10 11/ 33] *(645 /2205b) 0010 0x00A [*][X]
----- Объявление переменных -----
SECTION .data ; Секция иницированных данных
msg: DB 'Введите строку:',10 ; сообщение плюс
      символ перевода строки
msgLen: EQU $-msg ; Длина переменной 'msg'
SECTION .bss ; Секция не иницированных данных
buf1: RESB 80 ; Буфер размером 80 байт
архитектура ЭВМ

----- 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
1Помощь 2Сохран 3Блок 4Замена 5Копия 6Пер-ть 7Поиск 8Уда-ть 9МенюMC10Выход

```

Рис. 4.4: Файл lab05 в окне редактора mcedit

Просмотр содержимого файла *lab5-1.asm*(рис. 4.5).

```

/home/timurkarimov/work-ch-pc/lab05/lab5-1.asm      1409/2205      63%
----- Объявление переменных -----
SECTION .data ; Секция иницированных данных
msg: DB 'Введите строку:',10 ; сообщение плюс
      символ перевода строки
msgLen: EQU $-msg ; Длина переменной 'msg'
SECTION .bss ; Секция не иницированных данных
buf1: RESB 80 ; Буфер размером 80 байт
архитектура ЭВМ

----- 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
1Помощь 2Раз-рн 3Выход 4Нех 5Пер-ти 6 7Поиск 8Исх-ый 9Формат10Выход

```

Рис. 4.5: Выполнение команды F3 в mcedit

Оттранслируем исходный код из файла lab5-1.asm в объектный файл, затем выполним компоновку этого объектного файла и запустим его.(рис. 4.6).

```
timurkarimov@fedora:~/work/arch-pc/lab05$ nasm -f elf lab5-1.asm
timurkarimov@fedora:~/work/arch-pc/lab05$ ld -m elf_i386 -o lab5-1 lab5-1.o
timurkarimov@fedora:~/work/arch-pc/lab05$ ./lab5-1
Введите строку:
Каримов Тимур Ринатович
timurkarimov@fedora:~/work/arch-pc/lab05$
```

Рис. 4.6: Выполнение команды F3 в mcedit

Скачиваем файл in_out.asm со страницы курса ТУИС и копируем его в тот же каталог, что файл с программой.(рис. 4.7).

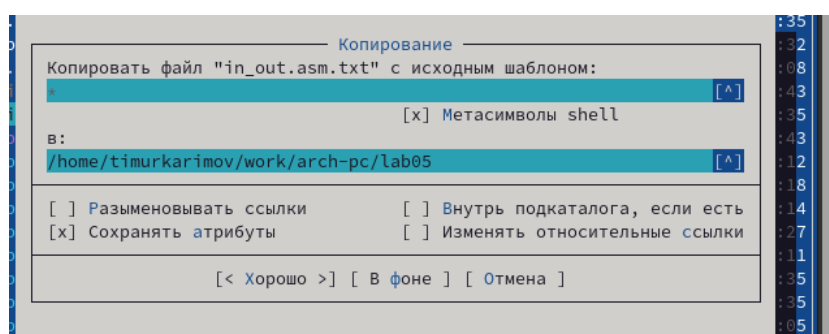


Рис. 4.7: Указание пути для копирования файла в mcedit

Исправим текст программы lab5-2.asm в соответствии с листингом 5.2(рис. 4.8) и проверим его работу(рис. 4.9)

```
%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 sprintfLF ; 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
```

Рис. 4.8: Исправление текст программы lab5-2.asm в соответствии с листингом 5.2

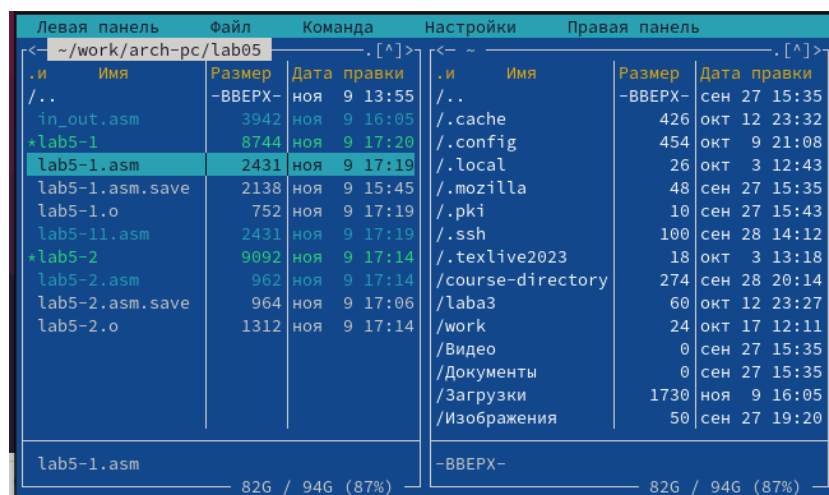


Рис. 4.12: Указание пути для копирования файла в mscdlt

Внесем изменения в соответствии с данным алгоритмом.(рис. 4.13).

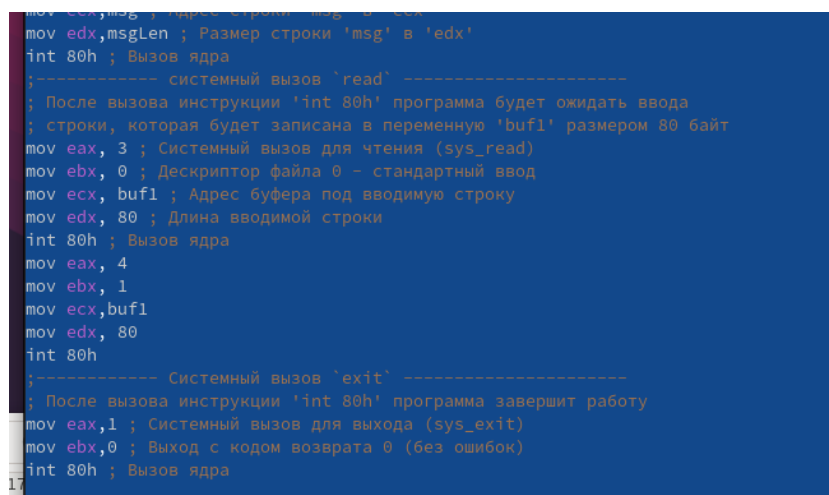


Рис. 4.13: Внесение изменения в соответствии с алгоритмом

Проверим работу созданного файла.(рис. 4.14).

```

Введите строку: cat lab5-1.o
timurkarimov@fedora:~/work/arch-pc/lab05$ touch lab5-1.asm
timurkarimov@fedora:~/work/arch-pc/lab05$ nasm -f elf lab5-1.asm
timurkarimov@fedora:~/work/arch-pc/lab05$ ld -m elf_i386 -o lab5-1 lab5-1.o
timurkarimov@fedora:~/work/arch-pc/lab05$ ./lab5-1
Введите строку:
sfjgdfg
timurkarimov@fedora:~/work/arch-pc/lab05$ mc
timurkarimov@fedora:~/work/arch-pc/lab05$ nasm -f elf lab5-11.asm
timurkarimov@fedora:~/work/arch-pc/lab05$ ld -m elf_i386 -o lab5-11 lab5-11.o
timurkarimov@fedora:~/work/arch-pc/lab05$ ./lab5-11
Введите строку:
Каримов Тимур Ринатович
Каримов Тимур Ринатович
timurkarimov@fedora:~/work/arch-pc/lab05$

```

Рис. 4.14: Запуск файла lab5-11.asm

Создадим копию файла lab5-2.asm под названием lab5-22.asm.(рис. 4.15).

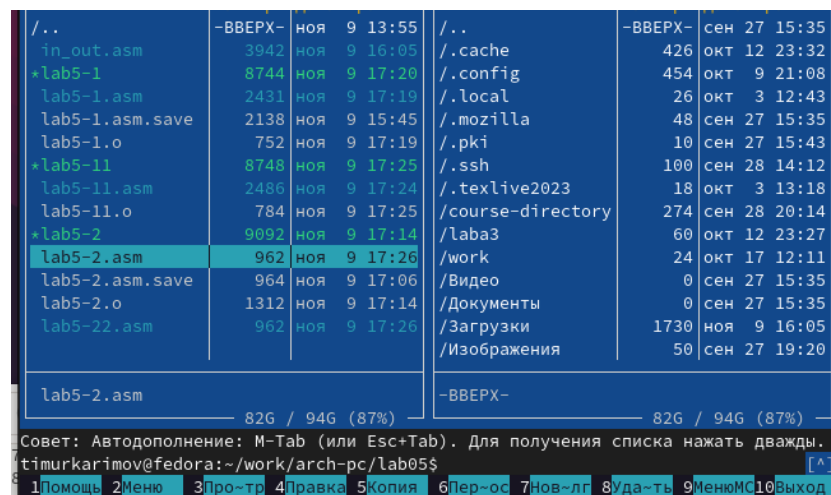


Рис. 4.15: Указание пути для копирования файла в mcedit

Внесем изменения в соответствии с данным алгоритмом.(рис. 4.16).

```
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
```

^G Справка ^O Записать ^W Поиск ^K Вырезать ^T Выполнить ^C Позиция
^X Выход ^R ЧитФайл ^\ Замена ^U Вставить ^J Выводить ^/ К строке

Рис. 4.16: Внесение изменения в соответствии с алгоритмом

Проверим работу созданного файла.(рис. 4.17).

```
timurkarimov@fedora:~/work/arch-pc/lab05$ nasm -f elf lab5-22.asm
timurkarimov@fedora:~/work/arch-pc/lab05$ ld -m elf_i386 -o lab5-22 lab5-22.o
timurkarimov@fedora:~/work/arch-pc/lab05$ ./lab5-22
Введите строку: Каримов Тимур Ринатович
Каримов Тимур Ринатович
timurkarimov@fedora:~/work/arch-pc/lab05$
```

Рис. 4.17: Запуск файла lab5-11.asm

5 Выводы

В ходе лабораторной работы студенты получили практические навыки работы с `ms` и изучили основы ассемблера `NASM`. Были приобретены навыки использования внешних файлов, значительно упрощающих написание кода.

Список литературы

1. Таненбаум Э., Бос Х. Современные операционные системы. 4-е изд. СПб.: Питер, 2015. 1120 с.
2. Robbins A. Bash Pocket Reference. O'Reilly Media, 2016. 156 с.
3. Zarrelli G. Mastering Bash. Packt Publishing, 2017. 502 с.
4. Newham C. Learning the bash Shell: Unix Shell Programming. O'Reilly Media, 2005. 354 с.