Отчёт по лабораторной работе №5

Дисциплина: архитектура компьютера.

Наговицын Арсений Владимирович

Содержание

# 1 Цель работы

Целью данной лабораторной работы является приобретение практических навыков работы в Midnight Commander, освоение инструкций языка ассемблера mov и int.

# 2 Задание

1. Основы работы с mc
2. Структура программы на языке ассемблера NASM
3. Подключение внешнего файла
4. Выполнение заданий для самостоятельной работы

# 3 Теоретическое введение

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. Программа на языке ассемблера NASM, как правило, состоит из трёх секций: секция кода программы (SECTION .text), секция инициированных (известных во время компиляции) данных (SECTION .data) и секция неинициализированных данных (тех, под которые во время компиляции только отводится память, а значение присваивается в ходе выполнения программы) (SECTION .bss). Для объявления инициированных данных в секции .data используются директивы DB, DW, DD, DQ и DT, которые резервируют память и указывают, какие значения должны храниться в этой памяти: - DB (define byte) — определяет переменную размером в 1 байт; - DW (define word) — определяет переменную размеров в 2 байта (слово); - DD (define double word) — определяет переменную размером в 4 байта (двойное слово); - DQ (define quad word) — определяет переменную размером в 8 байт (учетве- рённое слово); - DT (define ten bytes) — определяет переменную размером в 10 байт. Директивы используются для объявления простых переменных и для объявления массивов. Для определения строк принято использовать директиву DB в связи с особенностями хранения данных в оперативной памяти. Инструкция языка ассемблера mov предназначена для дублирования данных источника в приёмнике.

# 4 Выполнение лабораторной работы

## 4.1 Основы работы с mc

Открываю Midnight Commander, при помощи команды mc (рис. 4.1).

fig:
Меню mc

Рис. 4.1: Меню mc

Перехожу в каталог ~/work/study/2022-2023/Архитектура Компьютера/arch-pc (рис. 4.2).

fig:
Перемещение между директориями

Рис. 4.2: Перемещение между директориями

Перехожу в каталог ~/work/study/2022-2023/Архитектура Компьютера/arch-pc (рис. 4.3).

fig:
Перемещение между директориями

Рис. 4.2: Перемещение между директориями

С помощью функциональной клавиши F7 создаю каталог lab05 (рис. 4.3).

fig:
Создание каталога

Рис. 4.3: Создание каталога

Перехожу в созданный каталог (рис. 4.4).

fig:
Перемещение между директориями

Рис. 4.4: Перемещение между директориями

В строке ввода прописываю команду touch lab5-1.asm, чтобы создать файл, в котором буду работать (рис. 4.5).

fig:
Создание файла

Рис. 4.5: Создание файла

Проверяю создался ли файл (рис. 4.6).

fig:
Содержимое директории

Рис. 4.6: Содержимое директории

## 4.2 Структура программы на языке ассемблера NASM

С помощью функциональной клавиши F4 открываю созданный файл (рис. 4.7).

fig:
Содержимое файла

Рис. 4.7: Содержимое файла

Ввожу в файл код программы для запроса строки у пользователя (рис. 4.8).

fig:
Содержимое файла

Рис. 4.8: Содержимое файла

С помощью функциональной клавиши F3 открываю файл для просмотра, чтобы проверить, содержит ли файл текст программы (рис. 4.9).

fig:
Содержимое файла

Рис. 4.9: Содержимое файла

Транслирую текст программы файла в объектный файл (рис. 4.10).

fig:
Компиляция файла

Рис. 4.10: Компиляция файла

Проверяю правильность выполнения команды (рис. 4.11).

fig:
Содержимое директории

Рис. 4.11: Содержимое директории

Выполняю компоновку объектного файла (рис. 4.12).

fig:
Компиляция файла

Рис. 4.12: Компиляция файла

Проверяю правильность выполнения команды (рис. 4.13).

fig:
Содержимое директории

Рис. 4.13: Содержимое директории

Запускаю исполняемый файл. Программа выводит строку “Введите строку:” и ждет ввода с клавиатуры, я ввожу свои ФИО, на этом программа заканчивает свою работу (рис. 4.14).

fig:
Исполнение файла

Рис. 4.14: Исполнение файла

## 4.3 Подключение внешнего файла

Скачиваю файл in\_out.asm со страницы курса (рис. 4.15).

fig:
Содержимое директории

Рис. 4.15: Содержимое директории

С помощью функциональной клавиши F5 копирую файл in\_out.asm (рис. 4.16).

fig:
Копирование файла

Рис. 4.16: Копирование файла

С помощью функциональной клавиши F5 копирую файл lab6-1 в тот же каталог, но с другим именем (рис. 4.17).

fig:
Копирование файла

Рис. 4.17: Копирование файла

Проверяю правильность выполнения команды (рис. 4.18).

fig:
Содержимое директории

Рис. 4.18: Содержимое директории

Изменяю содержимое файла lab6-2.asm (рис. 4.19).

fig:
Редактирование файла

Рис. 4.19: Редактирование файла

Транслирую текст программы файла в объектный файл. Выполняю компоновку объектного файла. Запускаю исполняемый файл (рис. 4.20).

fig:
Исполнение файла

Рис. 4.20: Исполнение файла

Изменяю в файле подпрограмму sprintLF на sprint. Сохраняю изменения и открываю файл для просмотра, чтобы проверить сохранение действий (рис. 4.21).

fig:
Содержимое файла

Рис. 4.21: Содержимое файла

Снова транслирую файл, выполняю компоновку созданного объектного файла, запускаю новый исполняемый файл (рис. 4.22).

fig:
Исполнение файла

Рис. 4.22: Исполнение файла

Разница между первым исполняемым файлом и вторым заключается в том, что запуск первого запрашивает ввод с новой строки, а вторая программа, запрашивает ввод без переноса на новую строку.В этом заключается различие между подпрограммами sprintLF и sprint.

## 4.4 Выполнение заданий для самостоятельной работы

Создаю копию файла lab5-1.asm с именем lab5-1\_1.asm (рис. 4.23).

fig:
Копирование файла

Рис. 4.23: Копирование файла

Открываю созданный файл для редактирования. Изменяю программу так, чтобы кроме вывода приглашения и запроса ввода, она выводила вводимую пользователем строку (рис. 4.24).

fig:
Редактирование файла

Рис. 4.24: Редактирование файла

Создаю объектный файл, отдаю его на обработку компоновщику, получаю исполняемый файл , запускаю файл (рис. 4.25).

fig:
Исполнение файла

Рис. 4.25: Исполнение файла

Создаю копию файла lab5-2.asm с именем lab5-2\_1.asm (рис. 4.26).

fig:
Копирование файла

Рис. 4.26: Копирование файла

Открываю созданный файл для редактирования. Изменяю программу так, чтобы кроме вывода приглашения и запроса ввода, она выводила вводимую пользователем строку (рис. 4.27).

fig:
Редактирование файла

Рис. 4.27: Редактирование файла

Создаю объектный файл, отдаю его на обработку компоновщику, получаю исполняемый файл , запускаю файл (рис. 4.28).

fig:
Исполнение файла

Рис. 4.28: Исполнение файла

# 5 Выводы

При выполнении данной лабораторной работы я приобрел практические навыки работы в Midnight Commander, а также освоил инструкции языка ассемблера mov и int.