Отчёт по лабораторной работе №5

Архитектура компьютеров и операционные системы.

Брыляков Никита Евгеньевич

Содержание

# 1 Цель работы

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

# 2 Задание

1. Начало работы
2. Подключение внешнего файла in\_out.asm
3. 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 (учетверённое слово); • DT (define ten bytes) — определяет переменную размером в 10 байт Инструкция языка ассемблера intпредназначена для вызова прерывания с указанным номером. В общем виде она записывается в виде int n Здесь n — номер прерывания, принадлежащий диапазону 0–255. При программировании в Linux с использованием вызовов ядра sys\_calls n=80h (принято задавать в шестнадцатеричной системе счисления).

# 4 Выполнение лабораторной работы

## 4.1 Начало работы

Открываю Midnight Commander (рис. [1]).

fig:
Отрытие mc

Рис. 1: Отрытие mc

Перехожу в каталог ~/work/arch-pc созданный при выполнении лабораторной работы №4 (рис. [2]).

fig:
Переход в каталог

Рис. 2: Переход в каталог

Создаю папку lab05 и перехожу в созданный каталог. (рис. [3]).

fig:
Переход в созданный каталог

Рис. 3: Переход в созданный каталог

Создаю файл lab5-1.asm(рис. [4]).

fig:
Создание файла

Рис. 4: Создание файла

Открываю файл lab5-1.asm для редактирования во встроенном редакторе (рис. [5]).

fig:
Открытие файла

Рис. 5: Открытие файла

Ввожу текст программы и сохраняю (рис. [6]).

fig:
Ввод текста программы

Рис. 6: Ввод текста программы

Открываю файл lab5-1.asm для просмотра и убеждаюсь в наличии текста.(рис. [7]).

fig:
Просмотр файла

Рис. 7: Просмотр файла

Оттранслирую текст программы lab5-1.asm в объектный файл. Выполняю компоновку объектного файла и запускаю получившийся исполняемый файл. Программа выводит строку ‘Введите строку:’ и ожидает ввода с клавиатуры. На запрос ввожу своё ФИО (рис. [8]).

fig:
Компиляция и исполнение файла

Рис. 8: Компиляция и исполнение файла

## 4.2 Подключение внешнего файла in\_out.asm

Скачиваю файл in\_out.asm со страницы курса в ТУИС (рис. [9]).

fig:
Скачивание файла

Рис. 9: Скачивание файла

Копирую файл in\_out.asm из каталога Downloads в каталог lab05 (рис. [10]).

fig:
Копирование файла

Рис. 10: Копирование файла

Создаю копию файла lab5-1.asm с именем lab5-2.asm (рис. [11]).

fig:
Создание копии файла

Рис. 11: Создание копии файла

Изменяю содержимое файла lab-2.asm (рис. [12]).

fig:
Изменение программы

Рис. 12: Изменение программы

Создаю исполняемый файл и проверяю его работу. (рис. [13]).

fig:
Создание и проверка

Рис. 13: Создание и проверка

Открываю файл lab5-2.asm. Изменяю в нем подпрограмму sprintLF на sprint. (рис. [14]).

fig:
Изменение подпрограммы

Рис. 14: Изменение подпрограммы

Создаю исполняемый файл и проверяю его работу. (рис. [15]).

fig:
Создание и проверка

Рис. 15: Создание и проверка

Разница в том, что в случае с подпрограммой sprintLF исполняемый файл запрашивает ввод с новой строки, а с подпрогарммой sprint запрашивает ввод без переноса на новую строку

## 4.3 Выполнение заданий для самостоятельной работы

Создаю копию файла lab5-1.asm, называю её lab5-11.asm. (рис. [16]).

fig:
Копирование файла

Рис. 16: Копирование файла

Открываю программу и редактирую по заданию. (рис. [17]).

fig:
Редактирование

Рис. 17: Редактирование

Получаю исполняемый файл и проверяю его работу. (рис. [18]).

fig:
Проверка

Рис. 18: Проверка

Создаю копию файла lab5-2.asm, называю её lab5-22.asm. (рис. [19]).

fig:
Копирование файла

Рис. 19: Копирование файла

Открываю программу и редактирую по заданию. (рис. [20]).

fig:
Редактирование

Рис. 20: Редактирование

Получаю исполняемый файл и проверяю его работу. (рис. [21]).

fig:
Проверка

Рис. 21: Проверка

# 5 Вывод

При выполнении данной лабораторной работы я приобрёл практические навыки работы в Midnight Commander. Освоил инструкции языка ассемблера mov и int.

# 6 Список литературы

1. https://esystem.rudn.ru/pluginfile.php/2089085/mod\_resource/content/0/%D0%9B%D0%B0%D0%B1%D0%BE%D1%80%D0%B0%D1%82%D0%BE%D1%80%D0%BD%D0%B0%D1%8F%20%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%B0%20%E2%84%965.%20%D0%9E%D1%81%D0%BD%D0%BE%D0%B2%D1%8B%20%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D1%8B%20%D1%81%20Midnight%20Commander%20%28%29.%20%D0%A1%D1%82%D1%80%D1%83%D0%BA%D1%82%D1%83%D1%80%D0%B0%20%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D1%8B%20%D0%BD%D0%B0%20%D1%8F%D0%B7%D1%8B%D0%BA%D0%B5%20%D0%B0%D1%81%D1%81%D0%B5%D0%BC%D0%B1%D0%BB%D0%B5%D1%80%D0%B0%20NASM.%20%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%BD%D1%8B%D0%B5%20%D0%B2%D1%8B%D0%B7%D0%BE%D0%B2%D1%8B%20%D0%B2%20%D0%9E%D0%A1%20GNU%20Linux.pdf