

Отчет по лабораторной работе №5

Дисциплина: архитектура компьютера

Новиков Никита Владимирович

Содержание

1 Цель работы

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

2 Задание

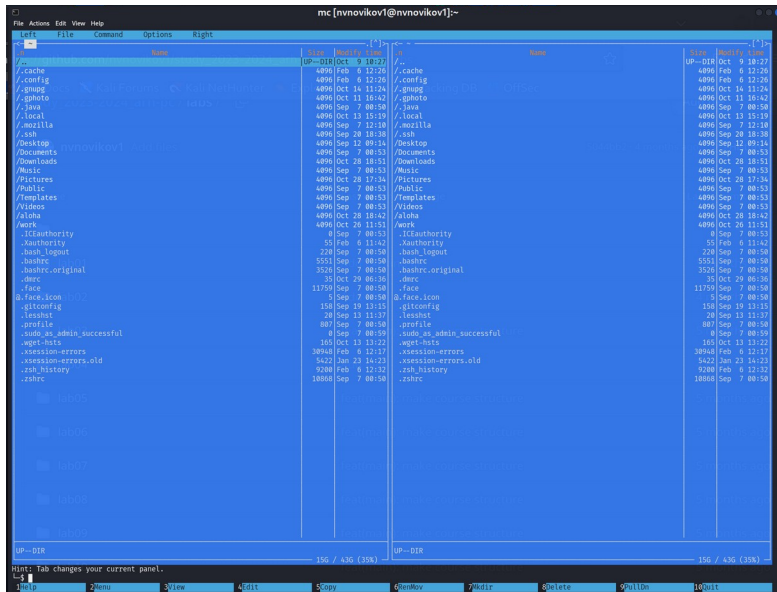
1. Основы работы с Midnight Commander
2. Структура программы на языке ассемблера NASM
3. Подключение внешнего файла

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`).

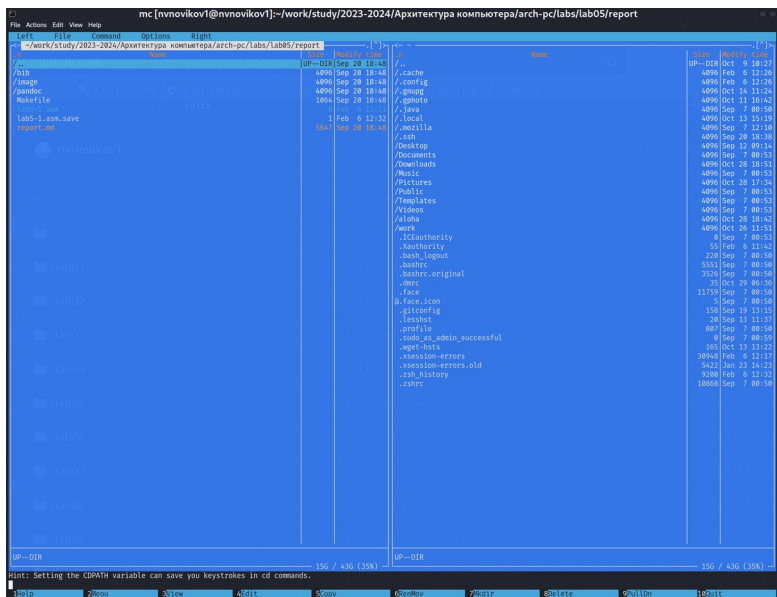
4 Выполнение лабораторной работы

Открываю Midnight Commander с помощью команды `mc`. (рис. [??]).



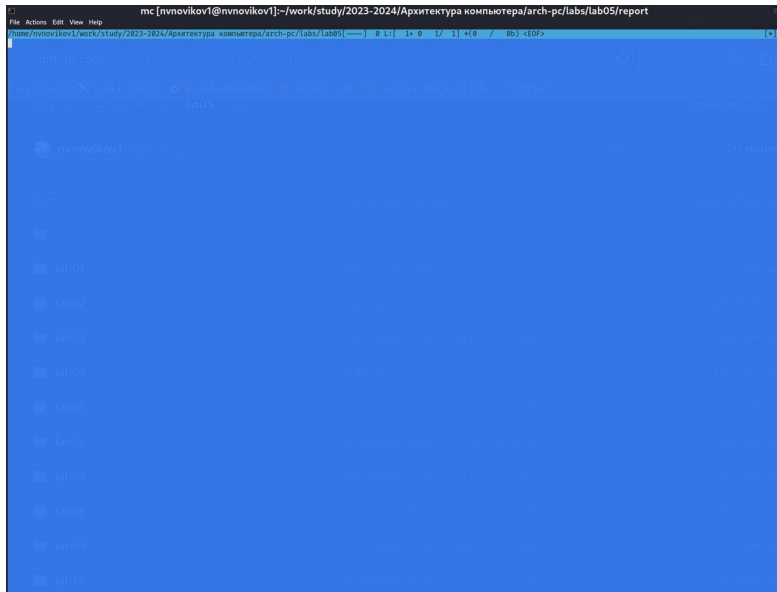
Открытые Midnight Commander

Пользуясь клавишами вверх, вниз и enter перехожу в каталог ~ /work/study/‘Архитектура компьютеров’/arch-pc/labs/lab05, созданный при выполнении прошлой лабораторной работы.(рис. [??]).



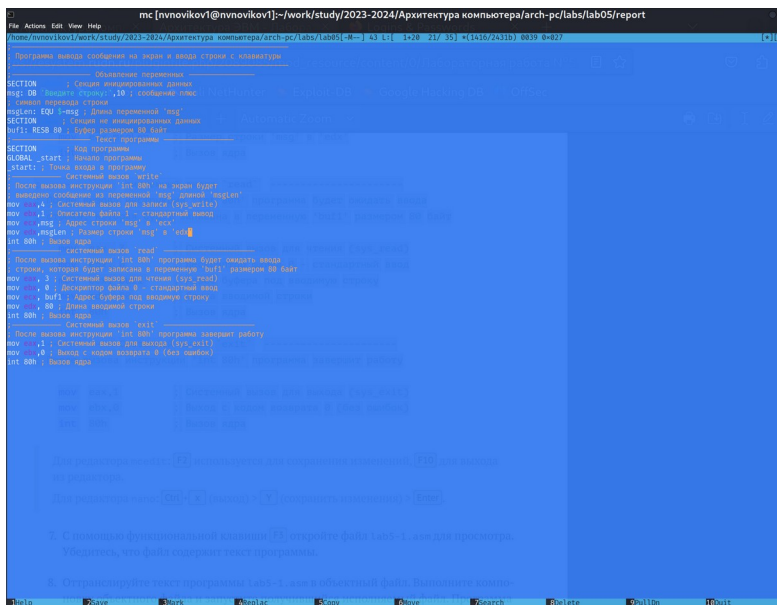
Переход в каталог

Пользуясь строкой ввода и командой touch создаю файл lab5-1.asm. С помощью функциональной клавиши F4 открываю файл lab05-1.asm для редактирования во встроенном редакторе.(рис. [??]).



Открытие редактора

Ввожу текст программы из листинга 5.1, сохраняю изменения и закрываю файл. (рис. [??]).



Ввод текста

Далее оттранслирую текст программы lab5-1.asm в объектный файл, выполняю компоновку объектного файла и запускаю получившийся исполняемый файл. Ввожу своё ФИО.(рис. [??]).

```

(nvnovikov1@nvnovikov1)-[~/../Архитектура компьютера/arch-pc/labs/lab05]
$ cd report

(nvnovikov1@nvnovikov1)-[~/../arch-pc/labs/lab05/report]
$ ls
Makefile  bib  image  lab5-1.asm  lab5-1.asm.save  pandoc  report.md

(nvnovikov1@nvnovikov1)-[~/../arch-pc/labs/lab05/report]
$ nasm -f elf lab5-1.asm

(nvnovikov1@nvnovikov1)-[~/../arch-pc/labs/lab05/report]
$ ld -m elf_i386 -o lab5-1 lab5-1.o

(nvnovikov1@nvnovikov1)-[~/../arch-pc/labs/lab05/report]
$ ./lab5-1
Введите строку:
Novikov Nikita Vladimirovich

```

Запуск файла

4.1 Подключение внешнего файла

Скачиваю файл `in_out.asm` со страницы курса в ТУИС. Копирую скаченный файл в директорию `lab05`. Создаю копию файла `lab5-1.asm` с именем `lab5-2.asm`. Исправляю текст программы в файле `lab5-2.asm` с использованием подпрограмм из внешнего файла `in_out.asm`. Далее создаю исполняемый файл и проверяю его работу.(рис. [??]).(рис. [??]).

```

File Actions Edit View Help
/home/nvnovikov1/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab05[None] 41 L: 1x16 1/7 1/7 *{1222/1222b} <EOF>
Программа вывода сообщений на экран в виде строки и столбца
include "in_out.asm" ; Подключение внешнего файла
SECTION
; Секция инициализированных данных
msg: DB "Введите строку: ",0h ; сообщение
SECTION
; Секция не инициализированных данных
buff: RESB 80 ; buffer размером 80 байт
SECTION
; Код программы
GLOBAL _start ; Начало программы
_start: ; Точка входа в программу
mov     %eax, msg ; записываем адрес входного сообщения в "EAX"
call    sprint ; вывод подпрограммы печати сообщения
mov     %eax, buff ; записываем адрес буфера в "EAX"
mov     %eax, 80 ; записываем длину входного сообщения в "EDX"
call    read ; вывод подпрограммы ввода сообщения
call    read ; вывод подпрограммы завершения
call    quit ; вывод подпрограммы завершения

(nvnovikov1@nvnovikov1)-[~/../arch-pc/labs/lab05/report]
$ nasm -f elf lab5-2.asm

(nvnovikov1@nvnovikov1)-[~/../arch-pc/labs/lab05/report]
$ ld -m elf_i386 -o lab5-2 lab5-2.o

(nvnovikov1@nvnovikov1)-[~/../arch-pc/labs/lab05/report]
$ ./lab5-2
Введите строку: Novikov Nikita Vladimirovich

```

5 Задание для самостоятельной работы

Создаю копию файла `lab5-1.asm` с названием `labсору.asm`. Далее редактирую файл `labсору.asm` так, чтобы она выводила то, что вводит пользователь.(рис. [??]).

```
/home/nvnovikov1/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab05/ 7 L: 1+34 35/ 40) *(2128/2459b) 0010 0x00A
; Программа вывода сообщения на экран и ввода строки с клавиатуры
; Объявление переменных
SECTION ; Секция инициализированных данных
msg: DB "Введите строку", 10 ; сообщение плюс
; символ перевода строки
msglen EQU $-msg ; длина переменной 'msg'
SECTION ; Секция не инициализированных данных
buf1: RESB 80 ; Буфер размером 80 байт
; Текст программы
SECTION ; 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 под вводимую строку
mov edx, 80 ; Длина вводимой строки
int 80h ; Вызов ядра
mov eax, 4
mov ebx, 1
mov ecx, buf1
mov edx, buf1
int 80h
; 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
```

Редактирование программы

Создаю объектный и исполняемый файлы и запускаю полученную программу. Ввожу свою фамилию и программа выводит мне её обратно. (рис. [??]).

```
(nvnovikov1@nvnovikov1)~[~/../arch-pc/labs/lab05/report]
$ nasm -f elf labcopy.asm

(nvnovikov1@nvnovikov1)~[~/../arch-pc/labs/lab05/report]
$ ld -m elf_i386 -o labcopy labcopy.o

(nvnovikov1@nvnovikov1)~[~/../arch-pc/labs/lab05/report]
$ ./labcopy
Введите строку: Nikita Novikov
Nikita Novikov
```

Проверка программы

После этого создаю другую копию файла lab5-2.asm и называю его labcopy2.asm. Также редактирую файл чтобы он выводил строку, введенную пользователем. (рис. [??]).

```
/home/nvnovikov1/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab05/ 7 L: 1+19 20/ 21) *(1194/1263b) 0010 0x00A
; Программа вывода сообщения на экран и ввода строки с клавиатуры
#include "labcopy.asm" ; подключение внешнего файла
SECTION ; Секция инициализированных данных
msg: DB "Введите строку", 0h ; сообщение
SECTION ; Секция не инициализированных данных
buf1: RESB 80 ; Буфер размером 80 байт
SECTION ; 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 ebx, buf1 ; запись адреса переменной в 'ebx'
mov ecx, 80 ; запись длины выводимого сообщения в 'ecx'
call sread ; вызов подпрограммы ввода сообщения
mov eax, 4
mov ebx, 1
mov ecx, buf1
mov edx, buf1
int 80h
call quit ; вызов подпрограммы завершения
```

Редактирование нового файла

После создаю объектный и исполняемый файлы для labcopy2.asm и запускаю эту программу. Также ввожу свою фамилию и получаю её обратно.(рис. [??]).

```
(nvnikov1@nvnikov1)-[~/.../arch-pc/labs/lab05/report]
$ nasm -f elf labcopy2.asm

(nvnikov1@nvnikov1)-[~/.../arch-pc/labs/lab05/report]
$ ld -m elf_i386 -o labcopy2 labcopy2.o

(nvnikov1@nvnikov1)-[~/.../arch-pc/labs/lab05/report]
$ ./labcopy2
Введите строку: nikita
nikita
```

Проверка работы программы

6 Выводы

После выполнения данной лабораторной работы я научился навыкам работы в Midnight Commnader и основе инструкция ассемблера mov и int.