

Лабораторная работа №6

Дисциплина: Архитектура компьютера

Александрова Ульяна Вадимовна

Содержание

1	Цель работы	5
2	Задание	6
3	Теоретическое введение	7
4	Выполнение лабораторной работы	8
5	Задачи для самостоятельной работы	12
6	Выводы	14

Список иллюстраций

4.1	Создание нового каталога	8
4.2	Создание файла	9
4.3	Текст программы	9
4.4	Запуск программы в терминале	10
4.5	Копирование подпрограммы в нужную папку	10
4.6	Создание файла программы	10
4.7	Запуск программы в терминале	11
4.8	Запуск измененной программы	11
5.1	Текст программы	12
5.2	Запущенная программа	12
5.3	Вывод измененной программы	13

Список таблиц

1 Цель работы

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

2 Задание

1. Освоение МС;
2. Заполнение отчета по выполнению Лабораторной работы №6;
3. Выполнение задания для самостоятельной работы;
4. Формирование выводов о проделанной работе.

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.

4 Выполнение лабораторной работы

Я открыла Midnight Commander в терминале и перешла в каталог arch-pc, а так же создала там папку lab06. (Рис. 4.1)

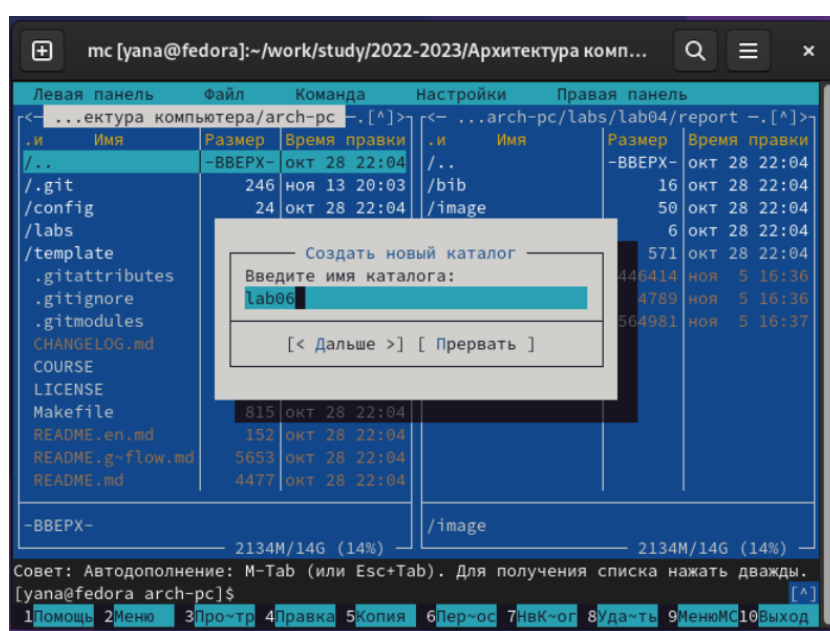


Рис. 4.1: Создание нового каталога

В созданной папке создала файл lab6-1.asm при помощи команды touch. (Рис. 4.2).


```
[yana@fedora lab06]$ ./lab6
Введите строку:
Александрова Ульяна
[yana@fedora lab06]$
```

Рис. 4.4: Запуск программы в терминале

В Midnight Commander скопировала файл in_out.asm в папку lab06 4.5.

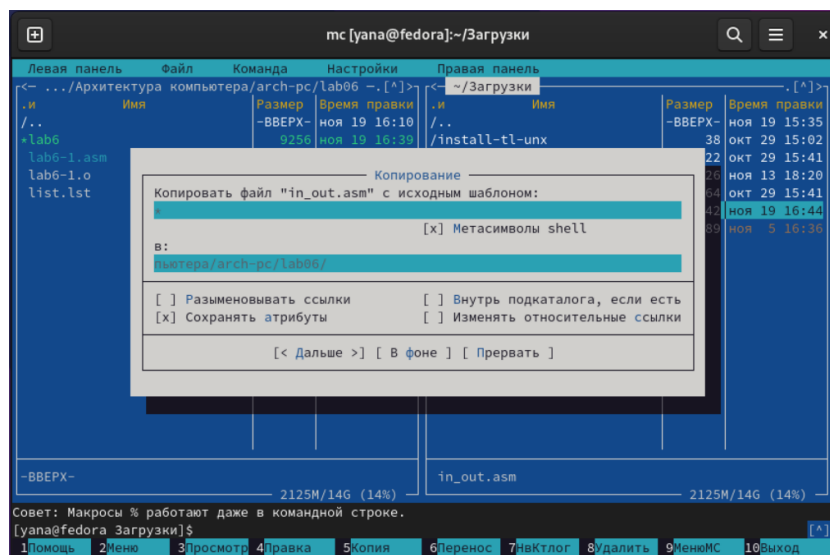


Рис. 4.5: Копирование подпрограммы в нужную папку

Затем создала файл lab6-02, скопировав предыдущий файл (Рис. 4.6).

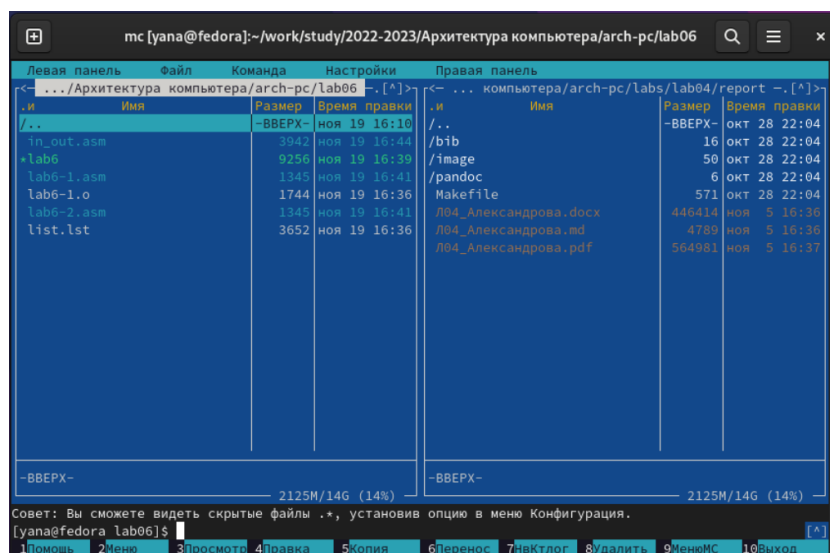


Рис. 4.6: Создание файла программы

Я использовала подпрограмму из файла `in_out.asm` в `lab6-2`, а потом проверила его работу, запустив (Рис. 4.7).

```
lab6-2.asm:18: error: symbol 'spread' not defined
[yana@fedora lab06]$ gedit lab6-2.asm
[yana@fedora lab06]$ nasm -f elf lab6-2.asm
[yana@fedora lab06]$ nasm -o lab6-2.o -f elf -g -l lab6-2.asm
lab6-2.o: fatal: no input file specified
Type nasm -h for help.
[yana@fedora lab06]$ nasm -o lab6-2.o -f elf -g -l list.lst lab6-2.asm
[yana@fedora lab06]$ ld -m elf_i386 lab6-2.o -o lab6-2
[yana@fedora lab06]$ ./lab6-2
Введите строку:
Александрова Ульяна
```

Рис. 4.7: Запуск программы в терминале

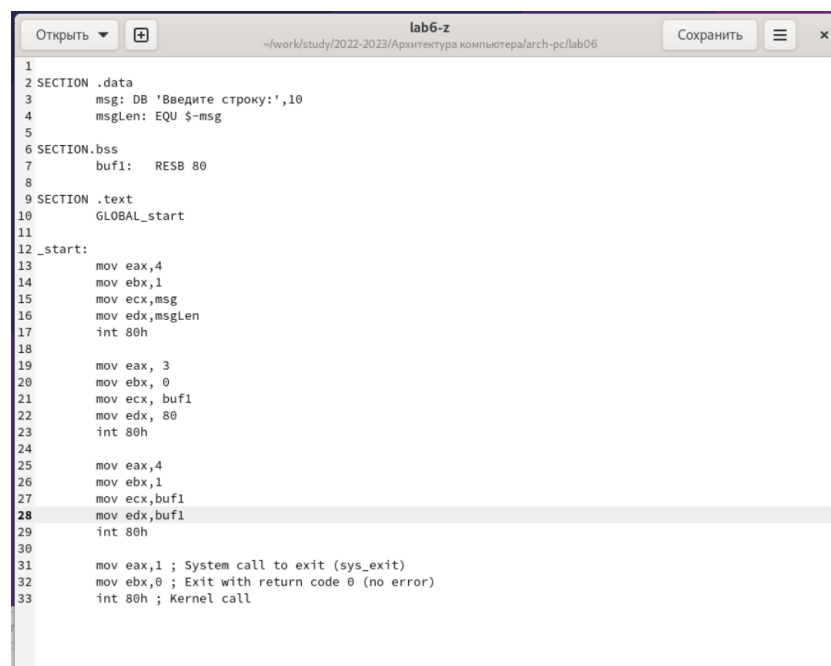
Изменив `sprintLF` на `sprint`, я выяснила, что это подпрограмма влияет на перенос текста (Рис. 4.8).

```
[yana@fedora lab06]$ ./lab6-2
Введите строку:
Александрова Ульяна
[yana@fedora lab06]$
```

Рис. 4.8: Запуск измененной программы

5 Задачи для самостоятельной работы

Я создала копию файла lab6-1.asm под названием lab6-z и внесла изменения в программу (без использования внешнего файла in_out.asm)(Рис. 5.1).



```
1
2 SECTION .data
3     msg: DB 'Введите строку:',10
4     msgLen: EQU $-msg
5
6 SECTION .bss
7     buf1: RESB 80
8
9 SECTION .text
10    GLOBAL _start
11
12 _start:
13     mov eax,4
14     mov ebx,1
15     mov ecx,msg
16     mov edx,msgLen
17     int 80h
18
19     mov eax,3
20     mov ebx,0
21     mov ecx,buf1
22     mov edx,80
23     int 80h
24
25     mov eax,4
26     mov ebx,1
27     mov ecx,buf1
28     mov edx,buf1
29     int 80h
30
31     mov eax,1 ; System call to exit (sys_exit)
32     mov ebx,0 ; Exit with return code 0 (no error)
33     int 80h ; Kernel call
```

Рис. 5.1: Текст программы

Причем так, чтобы она вывела введенную перед эти строку на экран (Рис. 5.2).



```
[yana@fedora lab06]$ ./lab6-z
Введите строку:
Александрова Ульяна
```

Рис. 5.2: Запущенная программа

Проделала то же самое, но с использованием подпрограммы (Рис. 5.3).

```
[yana@fedora lab06]$ ./lab6-z2  
Введите строку:Александрова Ульяна  
Александрова Ульяна  
[yana@fedora lab06]$
```

Рис. 5.3: Вывод измененной программы

6 Выводы

В результате проделанной работы, я приобрела практические навыки работы Midnight Commander, освоила инструкции языка ассемблера `mov` и `int`.