

Отчёт по лабораторной работе Архитектура компьютера

Лабораторная работа №4

Дмитрий Алексеевич Митяков

Содержание

1 Цель работы

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

2 Выполнение лабораторной работы

Я открыл командную строку и ввёл 'mc' и создал в нужном каталоге папку lab05 (рис. ??).

Левая панель	Файл	Команда
< ~/work/arch-pc . [^]>		
.и	Имя	Размер
-ВВЕРХ-		
офт 25 10:43		
/lab04	0	офт 25 10:43
/lab05	0	дек 15 00:57

Также строкой

ввода создал файл (рис. ??).

```
Совет: Автодополнение: M-Tab (или Esc+Tab). Для получения списка нажать дважды.
[damityakov@fedora lab05]$ touch lab5-1.asm
```

1Помощь	2Меню	3Про~тр	4Правка	5Копия	6Пер~ос	7Нов~лг	8Уда~ть	9МенюМС	10Выход
---------	-------	---------	---------	--------	---------	---------	---------	---------	---------

Скопировал и вставил нужный текст в файл (рис. ??).

```

mc [damityakov@fedora]:~/work/arch-pc/lab05
lab5-1.asm [----] 20 L: [ 1+ 8 9/ 35] *(607 /2431b) 1103 0x44F [*] [X]
;-----
; Программа вывода сообщения на экран и ввода строки с клавиатуры
;-----
;----- Объявление переменных -----
SECTION .data ; Секция инициализированных данных
msg: DB 'Введите строку:',10 ; сообщение плюс
; символ перевода строки
msglen: EQU $-msg ; Длина переменной 'msg'
SECTION .bss ; Секция не инициализированных данных
buf1: RESB 80 ; Буфер размером 80 байт
;----- 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
1Помощь 2Сохранить 3Блок 4Замена 5Копия 6Перейти 7Поиск 8Удалить 9МенюМС10Выход

```

Получил исполняемый файл и открыл его (рис. ??).

```

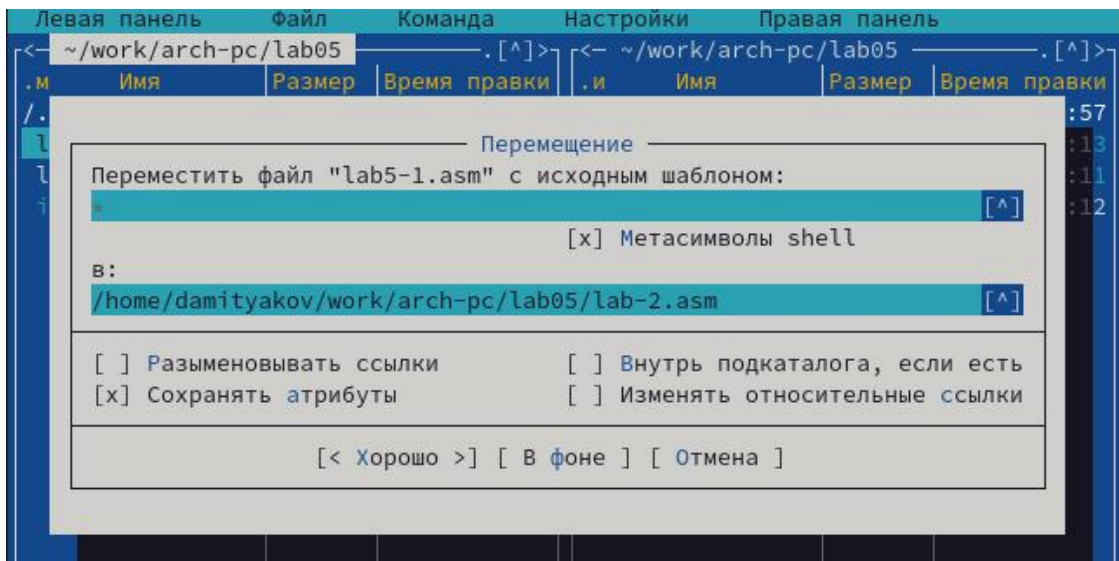
[damityakov@fedora lab05]$ nasm -f elf lab5-1.asm
[damityakov@fedora lab05]$ ld -m elf_i386 -o lab5-1 lab5-1.o
[damityakov@fedora lab05]$ ./lab5-1
Введите строку:
Митяков Дмитрий Алексеевич
[damityakov@fedora lab05]$

```

курса ТУИС скачал файл и скопировал его в директиву (рис. ??).

~ /work/arch-pc/lab05 . [^]>				~ /Загрузки . [^]>			
.м	Имя	Размер	Время правки	Имя	Размер	Время правки	
/..		-ВВЕРХ-	дек 15 00:57	/..		-ВВЕРХ-	дек 14 23:24
	lab5-1.asm	2431	дек 15 01:11		Лаборато~ASM.pdf	431971	дек 15 00:04
	lab5-1.o	752	дек 15 01:12		Л02_Митя~чет.pdf	544611	сен 20 19:31
*	lab5-1	8744	дек 15 01:12		Л02_Митя~(1).pdf	544611	сен 20 19:31
	in_out.asm	3942	дек 15 01:13		Л01_Митя~чет.pdf	717084	сен 20 19:38
					in_out.asm	3942	ноя 11 14:21
					fonstola~488.jpg	743211	сен 13 11:02

Создал копию файла(рис. ??).



Изменил файл заменив там код (рис. ??).

```
lab5-2.asm [----] 0 L: [ 1+17 18/ 18] *(1225/1225b) <EOF> [*][X]
;
; Программа вывода сообщения на экран и ввода строки с клавиатуры
;
%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 sprintf ; 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. ??).

```
[damityakov@fedora lab05]$ nasm -f elf lab5-2.asm
[damityakov@fedora lab05]$ ld -m elf_i386 -o lab5-2 lab5-2.o
[damityakov@fedora lab05]$ ./lab5-2
Введите строку:
Митяков Дмитрий Алексеевич
```

Заменял sprintf на printf, теперь ввод идет без переноса строки, в отличие от прошлого варианта(рис. ??).

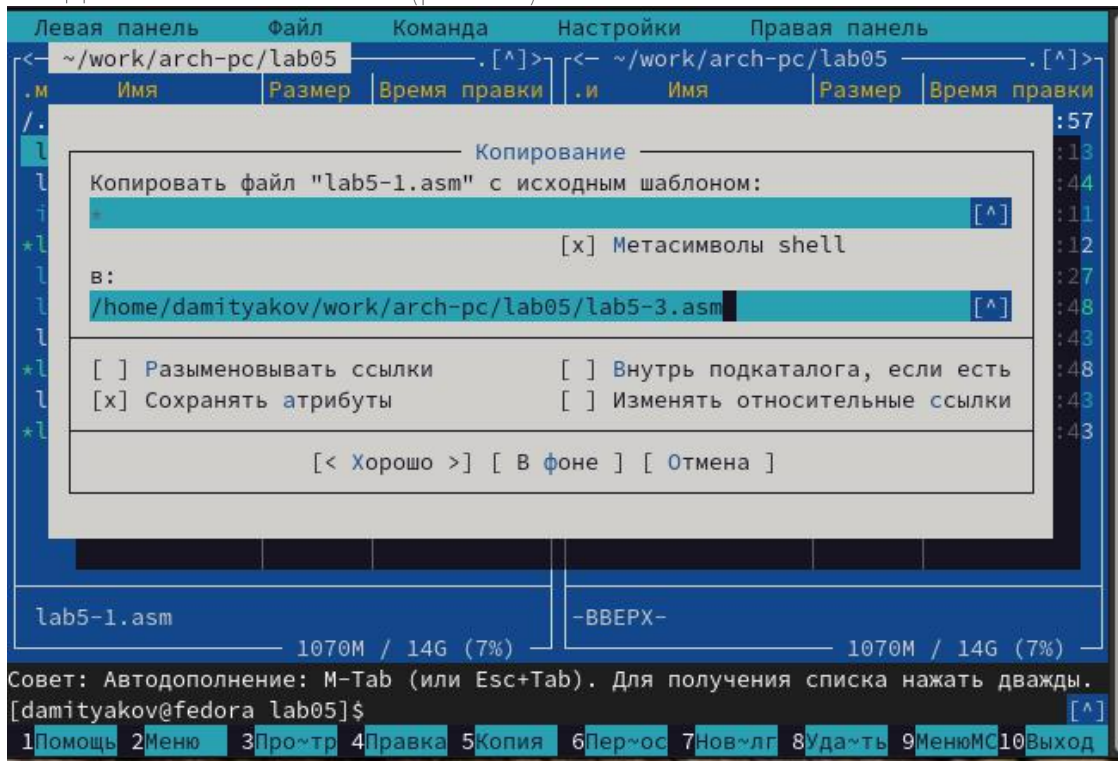
```
GLOBAL _start ; начало программы
_start: ; Точка входа в программу
mov eax, msg ; запись адреса выводимого сообщения в 'EAX'
call printf ; вызов подпрограммы печати сообщения
mov ecx, buf1 ; запись адреса переменной в 'EAX'
mov edx, 80 ; запись длины вводимого сообщения в 'EBX'
```

(рис. ??).

```
[damityakov@fedora lab05]$ nasm -f elf lab5-2-2.asm
[damityakov@fedora lab05]$ ld -m elf_i386 -o lab5-2-2 lab5-2-2.o
[damityakov@fedora lab05]$ ./lab5-2-2
Введите строку: Митяков Дмитрий Алексеевич
[damityakov@fedora lab05]$
```

3 Задание для самостоятельной работы

Создал копию lab5-1.asm(рис. ??).



И добавил туда необходимые строки, чтобы программа выводила содержимое переменной буф1 (рис. ??).


```

lab5-3.asm      [-----] 55 L:[ 22+10  32/ 42] *(2151/2731b) 0010 0x00A  [*][X]
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 ;Descriptor 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 ; Системный вызов для записи (sys_write)
mov ebx, 1 ; Описатель файла 1 - стандартный вывод
mov ecx, buf1 ; Адрес строки 'msg' в 'ecx'
mov edx, 80 ; Размер строки 'msg' в 'edx'
int 80h ; Вызов ядра
;----- 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

1Помощь 2Сохранить 3Блок 4Замена 5Копия 6Перейти 7Поиск 8Удалить 9МенюMC10Выход

```

(рис. ??).

```

Митяков
[damityakov@fedora lab05]$ nasm -f elf lab5-3.asm
[damityakov@fedora lab05]$ ld -m elf_i386 -o lab5-3 lab5-3.o
[damityakov@fedora lab05]$ ./lab5-3
Введите строку:
Митяков
Митяков

```

Аналогично сделал и для lab5-2.asm, здесь я добавил сохранение этой переменной в регистр eax и вызов функции sprint (рис. ??).

```

mov ecx, buf1 ; запись адреса переменной в `ECX`
mov edx, 80 ; запись длины вводимого сообщения в `EDX`
call sread ; вызов подпрограммы ввода сообщения
mov eax, buf1
call sprint
call quit ; вызов подпрограммы завершения

```

(рис. ??).

```

[damityakov@fedora lab05]$ nasm -f elf lab5-4.asm
[damityakov@fedora lab05]$ ld -m elf_i386 -o lab5-4 lab5-4.o
[damityakov@fedora lab05]$ ./lab5-4
Введите строку: Митяков
Митяков
[damityakov@fedora lab05]$

```

4 Выводы

Я приобрел практические навыки работы в Midnight Commander. Освоил инструкции языка ассемблера mov и int. Написал программу записи строки в переменную, сохранение и вывод.

Список литературы