

Лабораторная работа номер 4

Палымбетов Амирхан

Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение лабораторной работы	7
4	Выводы	11

Список иллюстраций

3.1	Рис1	7
3.2	Рис2	8
3.3	Рис3	8
3.4	Рис4	9
3.5	Рис5	10
3.6	Рис6	10

Список таблиц

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 тс
2. Подключение внешнего файла
3. Выполнение заданий для самостоятельной работы


```

/afs/.dk.sci.pfu.edu.ru/home/a/e/aepal~v/work/arch-pc/labs/lab05,
SECTION .data ; Секция иницированных данных
msg: DB 'Введите строку:',10
msgLen: EQU $-msg ; Длина переменной 'msg'
SECTION .bss ; Секция не иницированных данных
buf1: RESB 80 ; Буфер размером 80 байт
SECTION .text ; Код программы
GLOBAL _start ; Начало программы
_start: ; Точка входа в программу
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
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 ; Системный вызов для записи (sys_write)
mov ebx,1 ; Описатель файла '1' - стандартный вывод
mov ecx,buf1 ; Адрес строки buf1 в ecx
mov edx,buf1 ; Размер строки buf1
int 80h ; Вызов ядра
mov eax,1 ; Системный вызов для выхода (sys_exit)
mov ebx,0 ; Выход с кодом возврата 0 (без ошибок)
int 80h ; Вызов ядра

```

Рис. 3.4: Рис4

```

lab5-22.asm [----] 41 L: [ 1+17 18/ 18] *(1145/1145b) <E
%include 'in_out.asm'
SECTION .data ; Секция инициализированных данных
msg: DB "Введите строку: ",0h ; сообщение
SECTION .bss ; Секция не инициализированных 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 sprint ; 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
mov eax, 4 ; Системный вызов для записи (sys_write)
mov ebx, 1 ; Описатель файла '1' - стандартный вывод
mov ecx, buf1 ; Адрес строки buf1 в ecx
int 80h ; Вызов ядра
call quit ; вызов подпрограммы завершения

```

Рис. 3.5: Рис5

```

aepallhmbetov@dk8n64 ~/work/arch-pc/labs/lab05 $ ld -m elf_i386 lab5-22.o -o lab5-22
aepallhmbetov@dk8n64 ~/work/arch-pc/labs/lab05 $ nasm -o obj.o -f elf -g -l list.lst lab5-22.asm
aepallhmbetov@dk8n64 ~/work/arch-pc/labs/lab05 $ ld -m elf_i386 lab5-22.o -o lab5-22
aepallhmbetov@dk8n64 ~/work/arch-pc/labs/lab05 $ ./lab5-22
Введите строку: Палымбетов Амирхан
Палымбетов Амирхан
aepallhmbetov@dk8n64 ~/work/arch-pc/labs/lab05 $

```

Рис. 3.6: Рис6

4 Выводы

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