

Лабораторная работа №6

Дисциплина: Архитектура компьютера

Кондратьев Арсений Вячеславович

04.10.2022

Содержание

1	Цель работы	3
2	Выполнение лабораторной работы	4
3	Выводы	13
4	Контрольные вопросы	14

1 Цель работы

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

2 Выполнение лабораторной работы

1. С помощью функциональной клавиши F7 создайте папку lab06 и перейдите в созданный каталог(рис.2.1)

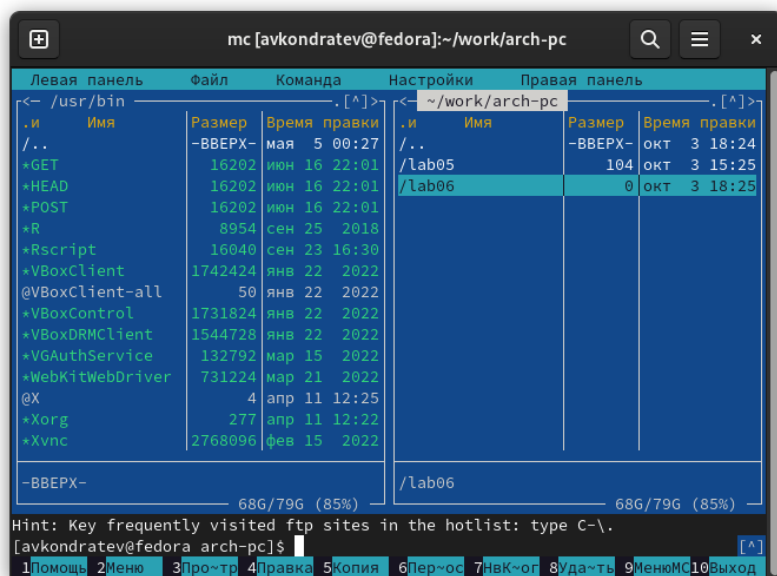


Figure 2.1: Рис. 1

2. Пользуясь строкой ввода и командой touch создайте файл lab6-1.asm(рис.2.2)

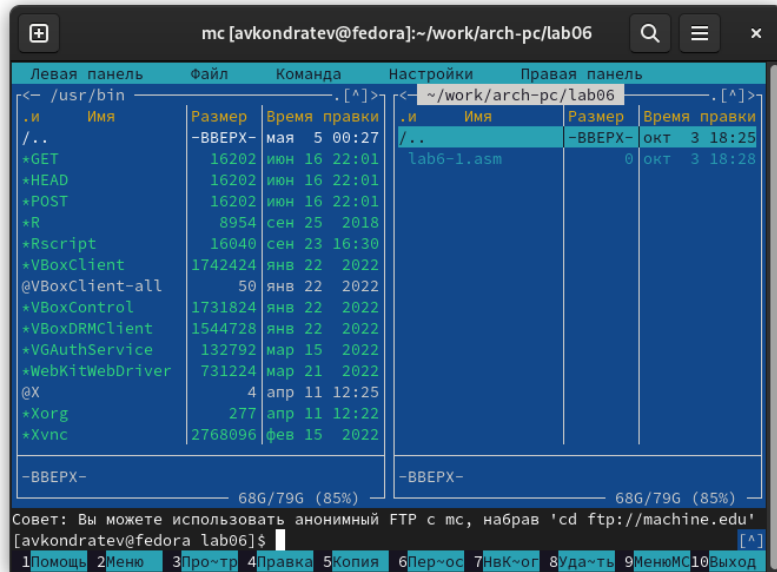


Figure 2.2: Рис. 2

3. С помощью функциональной клавиши F4 откройте файл lab6-1.asm для редактирования во встроенном редакторе(рис.2.3)

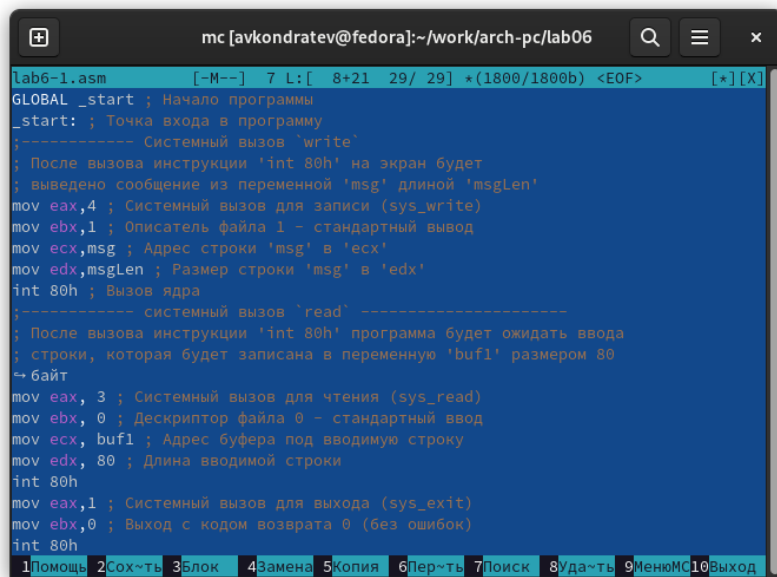


Figure 2.3: Рис. 3

4. Оттранслируйте текст программы lab6-1.asm в объектный файл. Выполните

компоновку объектного файла и запустите получившийся исполняемый файл(рис.2.4)

```
mc [avkondratev@fedora]:~/work/arch-pc/lab06
Подсчет объектов: 100% (24/24), готово.
Сжатие объектов: 100% (18/18), готово.
Запись объектов: 100% (18/18), 773.36 КиБ | 5.22 МиБ/с, готово.
Всего 18 (изменений 6), повторно использовано 0 (изменений 0), повторно использо-
вано пакетов 0
remote: Resolving deltas: 100% (6/6), completed with 2 local objects.
To github.com:ArsenyKondratev/study_2021-2022_arh-pc.git
 594377d..315e568 master -> master
[avkondratev@fedora arch-pc]$ cd
[avkondratev@fedora ~]$ mc
[avkondratev@fedora lab06]$ touch lab6.asm
[avkondratev@fedora lab06]$ touch lab6-1.asm
[avkondratev@fedora lab06]$ nasm -f elf lab6-1.asm
[avkondratev@fedora lab06]$ ld -m elf_i386 -o lab6-1 lab6-1.o
[avkondratev@fedora lab06]$ ./lab6-1
Введите строку:
Кондратьев Арсений Вячеславович
[avkondratev@fedora lab06]$
```

Figure 2.4: Рис. 4

5. Переместил in_out.asm в каталог с исполнительными файлами(рис.2.5)

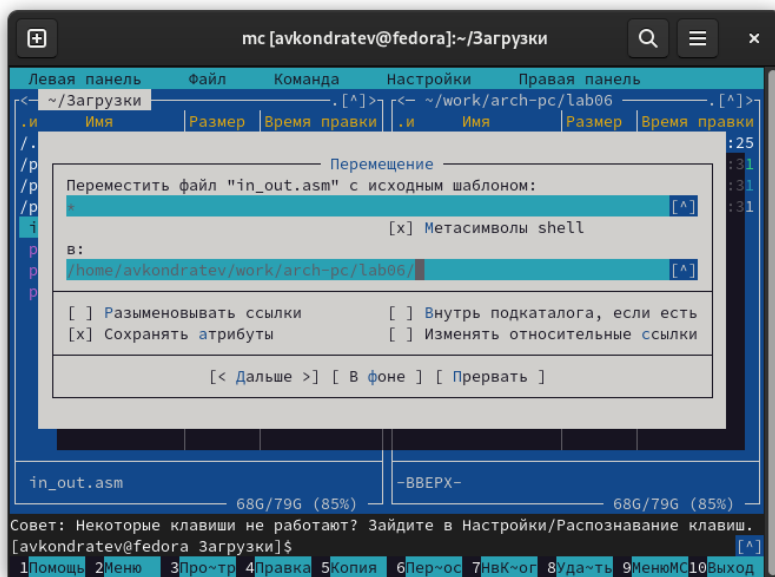


Figure 2.5: Рис. 5

6. С помощью функциональной клавиши F6 создайте копию файла lab6-1.asm с именем lab6-2.asm(рис.2.6)

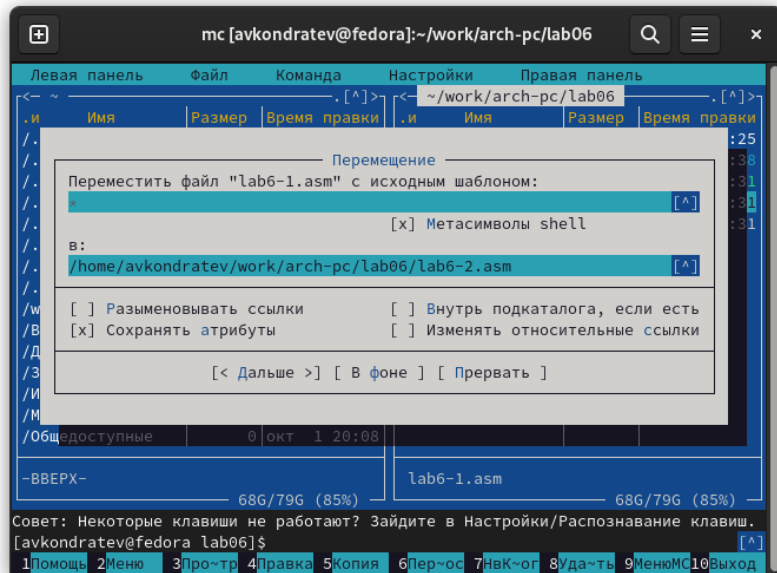
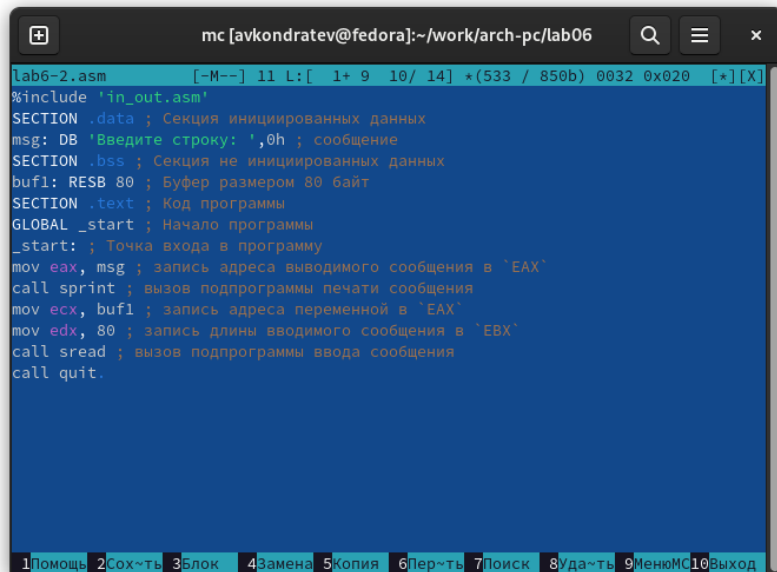


Figure 2.6: Рис. 6

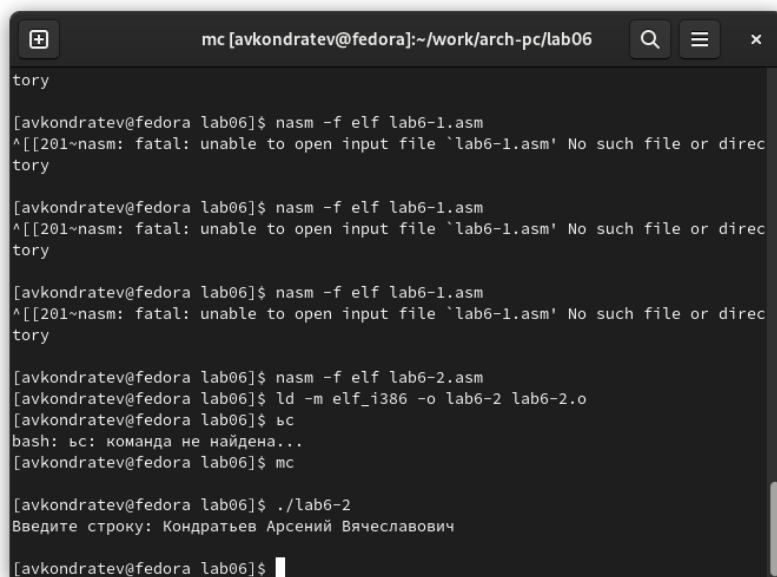
7. Исправьте текст программы в файле lab6-2.asm с использованием подпрограмм из внешнего файла in_out.asm(рис.2.7)



```
lab6-2.asm [-M--] 11 L: [ 1+ 9 10/ 14] *(533 / 850b) 0032 0x020 [*][X]
#include 'in_out.asm'
SECTION .data ; Секция иницированных данных
msg: DB 'Введите строку: ',0h ; сообщение
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, 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
call quit.
```

Figure 2.7: Рис. 7

8. В файле lab6-2.asm замените подпрограмму sprintLF на sprint. Создайте исполняемый файл и проверьте его работу(рис.2.8)



```
tory
[avkondratev@fedora lab06]$ nasm -f elf lab6-1.asm
^[[201~nasm: fatal: unable to open input file `lab6-1.asm' No such file or direc
tory

[avkondratev@fedora lab06]$ nasm -f elf lab6-1.asm
^[[201~nasm: fatal: unable to open input file `lab6-1.asm' No such file or direc
tory

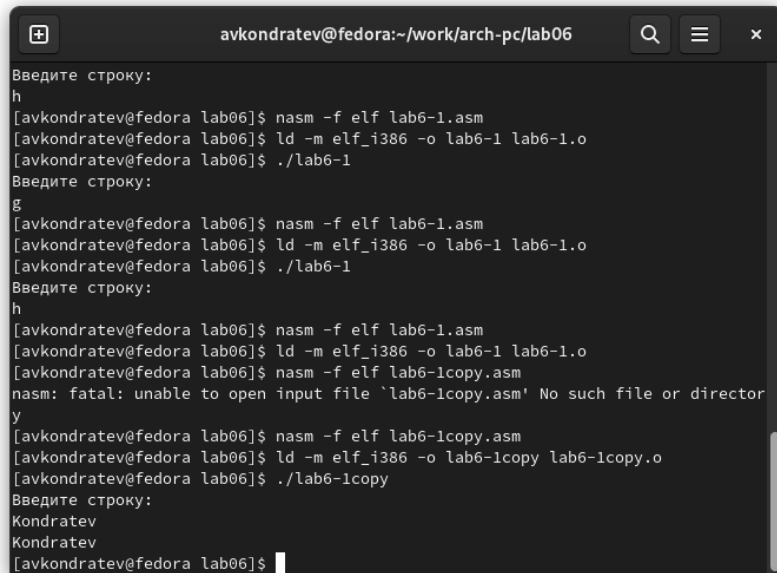
[avkondratev@fedora lab06]$ nasm -f elf lab6-1.asm
^[[201~nasm: fatal: unable to open input file `lab6-1.asm' No such file or direc
tory

[avkondratev@fedora lab06]$ nasm -f elf lab6-2.asm
[avkondratev@fedora lab06]$ ld -m elf_i386 -o lab6-2 lab6-2.o
[avkondratev@fedora lab06]$ ./lab6-2
Введите строку: Кондратьев Арсений Вячеславович
[avkondratev@fedora lab06]$
```

Figure 2.8: Рис. 8

Разница в том, что sprintLF переносит строку, а sprint нет.

9. Создайте копию файла lab6-1.asm. Внесите изменения в программу (без использования внешнего файла in_out.asm), так чтобы она выводила введенный текст(рис.2.9)(рис.2.10)



```
avkondratev@fedora:~/work/arch-pc/lab06
Введите строку:
h
[avkondratev@fedora lab06]$ nasm -f elf lab6-1.asm
[avkondratev@fedora lab06]$ ld -m elf_i386 -o lab6-1 lab6-1.o
[avkondratev@fedora lab06]$ ./lab6-1
Введите строку:
g
[avkondratev@fedora lab06]$ nasm -f elf lab6-1.asm
[avkondratev@fedora lab06]$ ld -m elf_i386 -o lab6-1 lab6-1.o
[avkondratev@fedora lab06]$ ./lab6-1
Введите строку:
h
[avkondratev@fedora lab06]$ nasm -f elf lab6-1.asm
[avkondratev@fedora lab06]$ ld -m elf_i386 -o lab6-1 lab6-1.o
[avkondratev@fedora lab06]$ nasm -f elf lab6-1copy.asm
nasm: fatal: unable to open input file 'lab6-1copy.asm' No such file or director
y
[avkondratev@fedora lab06]$ nasm -f elf lab6-1copy.asm
[avkondratev@fedora lab06]$ ld -m elf_i386 -o lab6-1copy lab6-1copy.o
[avkondratev@fedora lab06]$ ./lab6-1copy
Введите строку:
Kondratev
Kondratev
[avkondratev@fedora lab06]$
```

Figure 2.9: Результат

```
lab6-1copy.asm
~\work\arch-pc\lab06

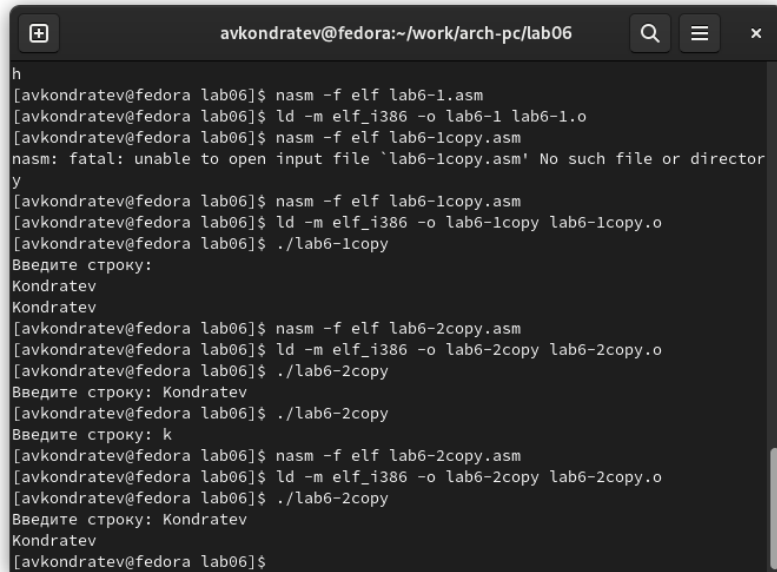
SECTION .data ; Секция иницированных данных
msg: DB 'Введите строку:',10
msgLen: EQU $-msg ; Длина переменной 'msg'
SECTION .bss ; Секция не иницированных данных
buf1: RESB 80 ; Буфер размером 80 байт
;----- Текст программы -----
SECTION .text ; Код программы
GLOBAL _start ; Начало программы
_start: ; 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
;----- системный вызов 'read' -----
; После вызова инструкции 'int 80h' программа будет ожидать ввода
; строки, которая будет записана в переменную 'buf1' размером 80
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

mov eax,4
mov ebx,1
mov ecx,buf1
mov edx,80
int 80h

mov eax,1 ; Системный вызов для выхода (sys_exit)
mov ebx,0 ; Выход с кодом возврата 0 (без ошибок)
int 80h
```

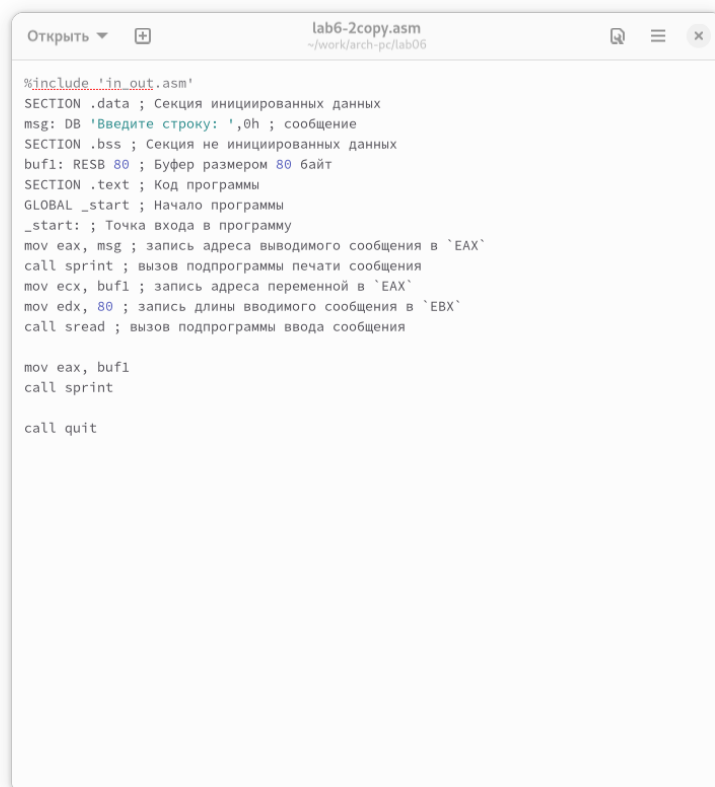
Figure 2.10: Код

10. Создайте копию файла lab6-2.asm. Исправьте текст программы с использование подпрограмм из внешнего файла in_out.asm, так чтобы она выводила введенный текст(рис.2.9)(рис.2.10)



```
h
[avkondratev@fedora lab06]$ nasm -f elf lab6-1.asm
[avkondratev@fedora lab06]$ ld -m elf_i386 -o lab6-1 lab6-1.o
[avkondratev@fedora lab06]$ nasm -f elf lab6-1copy.asm
nasm: fatal: unable to open input file `lab6-1copy.asm' No such file or director
y
[avkondratev@fedora lab06]$ nasm -f elf lab6-1copy.asm
[avkondratev@fedora lab06]$ ld -m elf_i386 -o lab6-1copy lab6-1copy.o
[avkondratev@fedora lab06]$ ./lab6-1copy
Введите строку:
Kondratev
Kondratev
[avkondratev@fedora lab06]$ nasm -f elf lab6-2copy.asm
[avkondratev@fedora lab06]$ ld -m elf_i386 -o lab6-2copy lab6-2copy.o
[avkondratev@fedora lab06]$ ./lab6-2copy
Введите строку: Kondratev
[avkondratev@fedora lab06]$ ./lab6-2copy
Введите строку: k
[avkondratev@fedora lab06]$ nasm -f elf lab6-2copy.asm
[avkondratev@fedora lab06]$ ld -m elf_i386 -o lab6-2copy lab6-2copy.o
[avkondratev@fedora lab06]$ ./lab6-2copy
Введите строку: Kondratev
Kondratev
[avkondratev@fedora lab06]$
```

Figure 2.11: Результат







```
Открыть ▾  lab6-2copy.asm  
~\work\arch-pc\lab06     
  
%include 'in_out.asm'  
SECTION .data ; Секция иницированных данных  
msg: DB 'Введите строку: ',0h ; сообщение  
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, 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, buf1  
call sprint  
  
call quit
```

Figure 2.12: Код

3 Выводы

Я приобрел практические навыки работы в Midnight Commander. Освоил инструкции языка ассемблера `mov` и `int`.

4 Контрольные вопросы

1. 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
2. Копирование(cp; F5)
перенос(mv; F6)
просмотр(cat; F3)
создание каталога(mkdir; F7)
удаление(rm; F8)
3. состоит из трёх секций: секция кода программы (SECTION .text), секция инициализированных (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. неинициализированных и инициализированных
5. DB (define byte) — определяет переменную размером в 1 байт;
DW (define word) — определяет переменную размером 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 (учетве-

рѐнное слово);

DT (define ten bytes) — определяет переменную размером в 10 байт.

6. произойдет дублирование данных esi в eax

7. Вызов ядра