

Отчёт по лабораторной работе №5

**Основы работы с Midnight Commander (mc). Структура
программы на языке ассемблера NASM. Системные вызовы в ОС
GNU Linux**

Лань Цяньин

1. Цель работы

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

2. Порядок выполнения лабораторной работы

2.1. Запуск Midnight Commander и переход в рабочий каталог

Запуск Midnight Commander командой `mc` и переход в каталог `~/work/arch-
pc` с помощью клавиш `↑/↓` и `Enter` (рис. Рисунок 1).

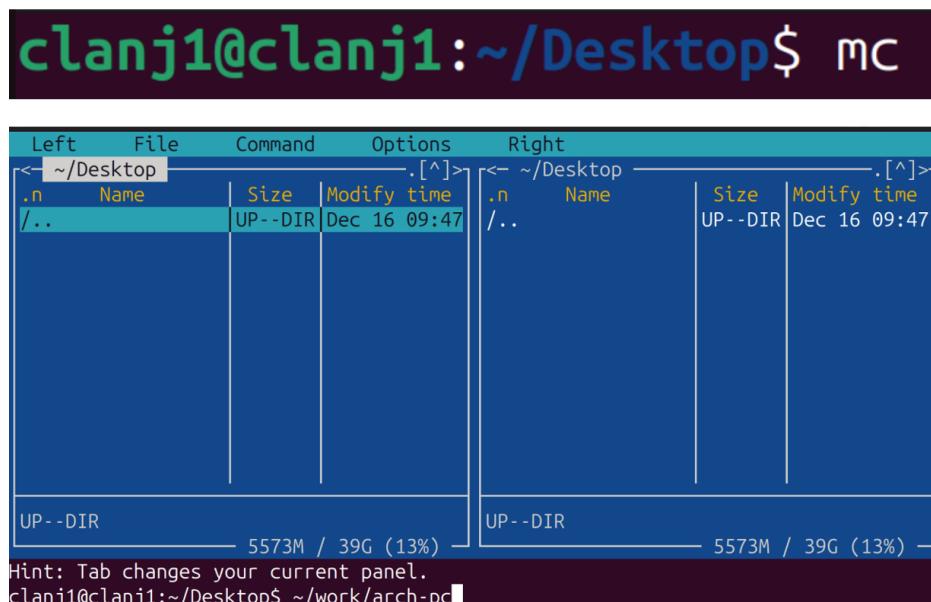


Рисунок 1: Запуск `mc` и переход в `~/work/arch-
pc`

Открытие `mc` и переход в каталог `~/work/arch-
pc` выполнены успешно.

2.2. Создание каталога для лабораторной работы и файла программы

С помощью функциональной клавиши **F7** создана папка lab05 и выполнен переход в созданный каталог; затем с помощью строки ввода и команды `touch` создан файл `lab5-1.asm` (рис. Рисунок 2).

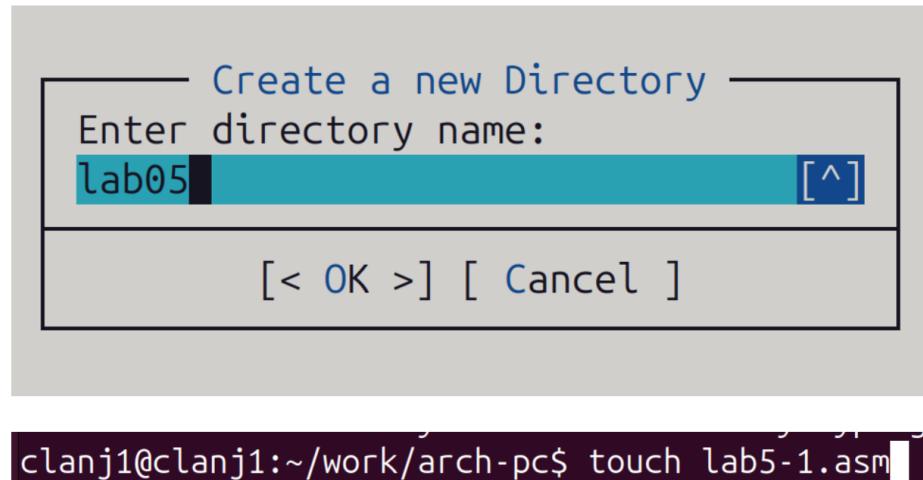
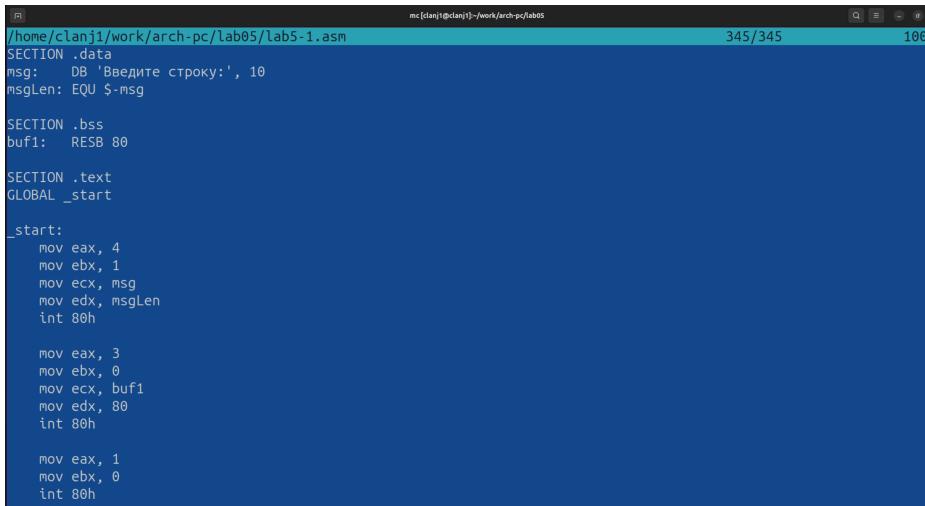


Рисунок 2: Создание каталога lab05 и файла lab5-1.asm

В результате был создан каталог `lab05`, а также файл `lab5-1.asm` для дальнейшего редактирования программы.

2.3. Редактирование и ввод первой программы на ассемблере

Открытие файла `lab5-1.asm` клавишей **F4** во встроенным редакторе `Midnight Commander` и ввод текста программы (рис. Рисунок 3).



```
/home/clanj1/work/arch-pc/lab05/lab5-1.asm
SECTION .data
msg: DB 'Введите строку:', 10
msgLen: EQU $-msg

SECTION .bss
buf1: RESB 80

SECTION .text
GLOBAL _start

_start:
    mov eax, 4
    mov ebx, 1
    mov ecx, msg
    mov edx, msgLen
    int 80h

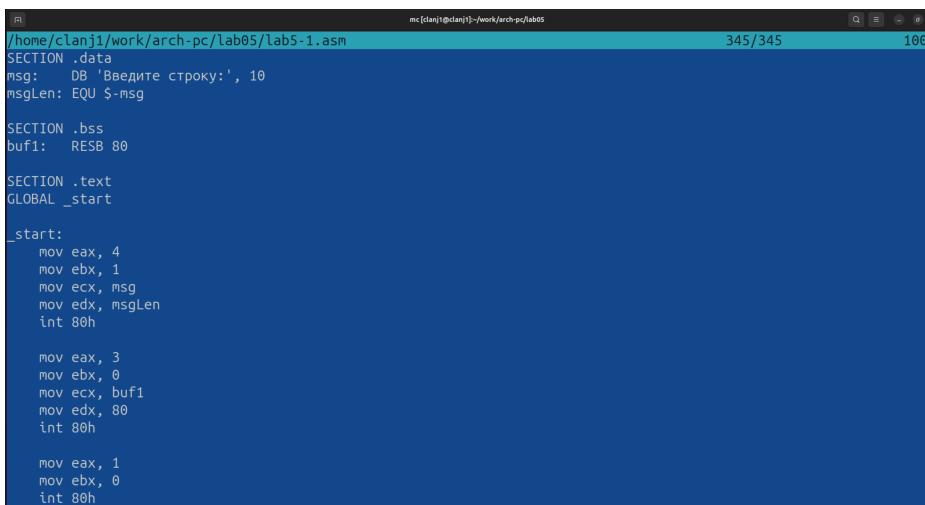
    mov eax, 3
    mov ebx, 0
    mov ecx, buf1
    mov edx, 80
    int 80h

    mov eax, 1
    mov ebx, 0
    int 80h
```

Рисунок 3: Редактирование файла lab5-1.asm во встроенным редакторе Midnight Commander

Файл lab5-1.asm успешно открыт и отредактирован во встроенным редакторе.

—
Файл lab5-1.asm был открыт для просмотра с помощью клавиши **F3** с целью проверки его содержимого (рис. Рисунок 4).



```
/home/clanj1/work/arch-pc/lab05/lab5-1.asm
SECTION .data
msg: DB 'Введите строку:', 10
msgLen: EQU $-msg

SECTION .bss
buf1: RESB 80

SECTION .text
GLOBAL _start

_start:
    mov eax, 4
    mov ebx, 1
    mov ecx, msg
    mov edx, msgLen
    int 80h

    mov eax, 3
    mov ebx, 0
    mov ecx, buf1
    mov edx, 80
    int 80h

    mov eax, 1
    mov ebx, 0
    int 80h
```

Рисунок 4: Просмотр файла lab5-1.asm

Код отображается, значит файл создан и заполнен правильно

2.4. Трансляция, компоновка и запуск программы

lab5-1

Выполнена трансляция программы lab5-1.asm в объектный файл, компоновка и запуск полученного исполняемого файла (рис. Рисунок 5).

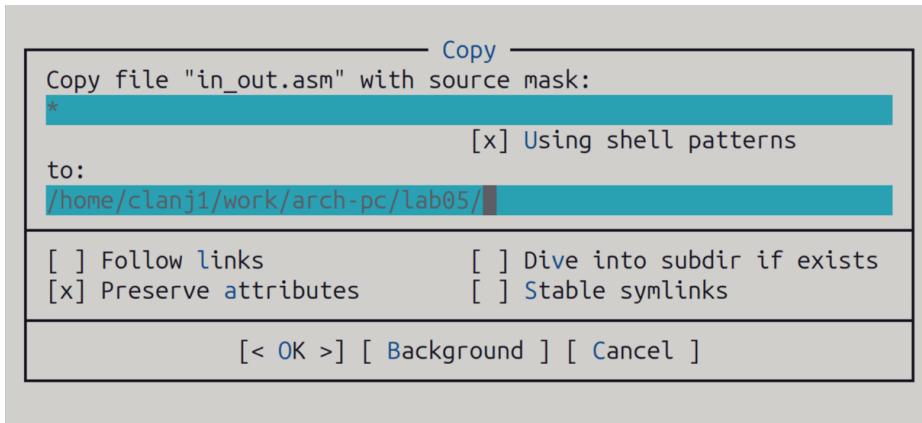
```
clanj1@clanj1:~/work/arch-pc/lab05$ nasm -f elf lab5-1.asm  
clanj1@clanj1:~/work/arch-pc/lab05$ ld -m elf_i386 -o lab5-1 lab5-1.o  
clanj1@clanj1:~/work/arch-pc/lab05$ ./lab5-1  
Введите строку:  
Лань Цяньин
```

Рисунок 5: Трансляция, компоновка и запуск программы

Программа выводит приглашение и принимает ввод строки, что подтверждает её корректную работу.

2.5. Подключение внешнего файла in_out.asm и создание второй программы

Копирование файла in_out.asm в каталог lab05 (к файлу lab5-1.asm) с помощью клавиши **F5** (рис. Рисунок 6).



'*in_out.asm	3942	Dec 16 11:21
*lab5-1	8744	Dec 16 11:10
lab5-1.asm	345	Dec 16 10:06
lab5-1.o	752	Dec 16 10:13

Рисунок 6: Копирование in_out.asm в lab05

Окно копирования показывает назначение .../lab05 и файл in_out.asm; после подтверждения он будет лежать рядом с lab5-1.asm.

—
Создание копии файла lab5-1.asm с именем lab5-2.asm с помощью функциональной клавиши **F6** (рис. Рисунок 7).

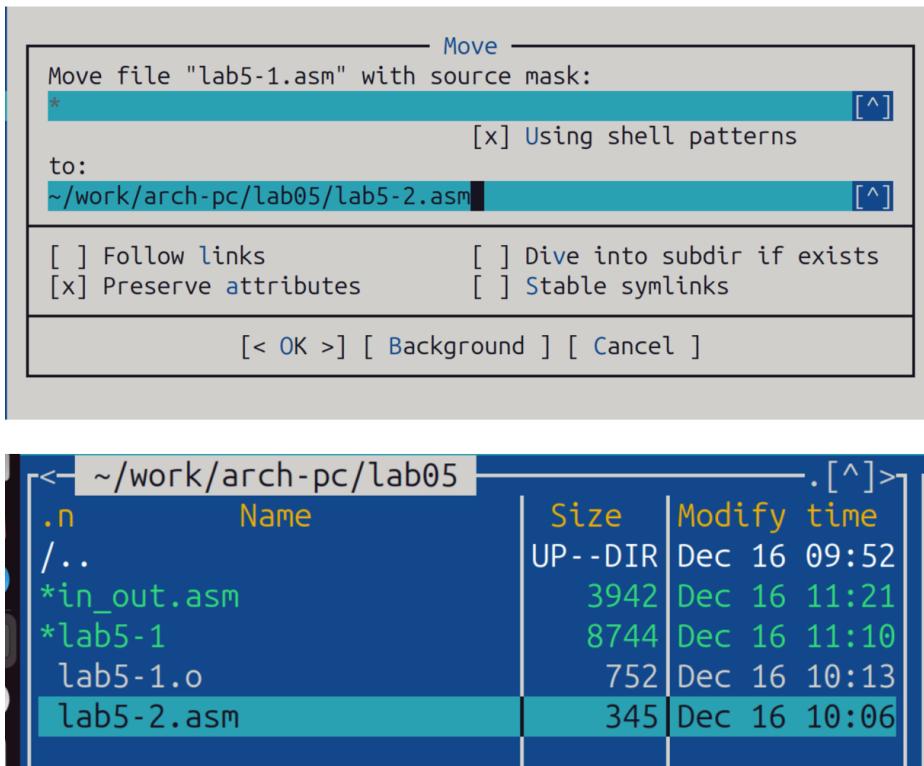


Рисунок 7: Создание копии файла lab5-2.asm

Появление файла `lab5-2.asm` в каталоге подтверждает успешное создание копии.

Исправление программы в файле `lab5-2.asm`: подключение внешнего файла `in_out.asm` и замена ввода/вывода на подпрограммы `sprintLF`, `sread`, `quit` по листингу 5.2 (рис. Рисунок 8).

```
GNU nano 7.2          /home/clanj1/work/arch-pc/lab05/lab5-2.asm
%include 'in_out.asm'

SECTION .data
msg: DB 'Введите строку: ',0h

SECTION .bss
buf1: RESB 80

SECTION .text
GLOBAL _start

_start:

    mov eax, msg
    call sprintLF

    mov ecx, buf1
    mov edx, 80
    call sread
```

Рисунок 8: Редактирование lab5-2.asm и подключение in_out.asm

Окно редактора и наличие директивы `%include` вместе с вызовами `sprintLF/sread` подтверждают, что файл `lab5-2.asm` исправлен по листингу 5.2 и готов к сборке и проверке.

—
Сборка и запуск исправленной программы `lab5-2.asm`: трансляция в объектный файл, компоновка и выполнение(рис. Рисунок 9).

```
clanj1@clanj1:~/work/arch-pc/lab05$ nasm -f elf lab5-2.asm
clanj1@clanj1:~/work/arch-pc/lab05$ ld -m elf_i386 -o lab5-2 lab5-2.o
clanj1@clanj1:~/work/arch-pc/lab05$ ./lab5-2
Введите строку:
Лань
```

Рисунок 9: Сборка и запуск программы lab5-2

Появление приглашения «Введите строку» и корректный ввод подтверждают, что программа успешно собрана и работает правильно.

2.6. Проверка различий между подпрограммами **sprintLF** и **sprint**

Замена подпрограммы `sprintLF` на `sprint` в файле `lab5-2.asm`, повторная сборка и запуск программы.(рис. Рисунок 10).



The screenshot shows a terminal window with the following content:

```
_start:  
    mov eax, msg  
    call sprint  
}  
  
nasm -f elf lab5-2.asm  
clanj1@clanj1:~/work/arch-pc/lab05$ ld -m elf_i386 -o lab5-2 lab5-2.o  
clanj1@clanj1:~/work/arch-pc/lab05$ ./lab5-2  
Введите строку: Лань
```

Рисунок 10: Замена `sprintLF` на `sprint` и запуск программы

Выводимый текст не переходит автоматически на новую строку, что показывает, что подпрограмма `sprint` не содержит символа перевода строки, тогда как `sprintLF` выводит перевод строки в конце

3. Задание для самостоятельной работы

Создание копии файла lab5-1.asm с именем lab5-1-ind.asm.(рис. Рисунок 1).

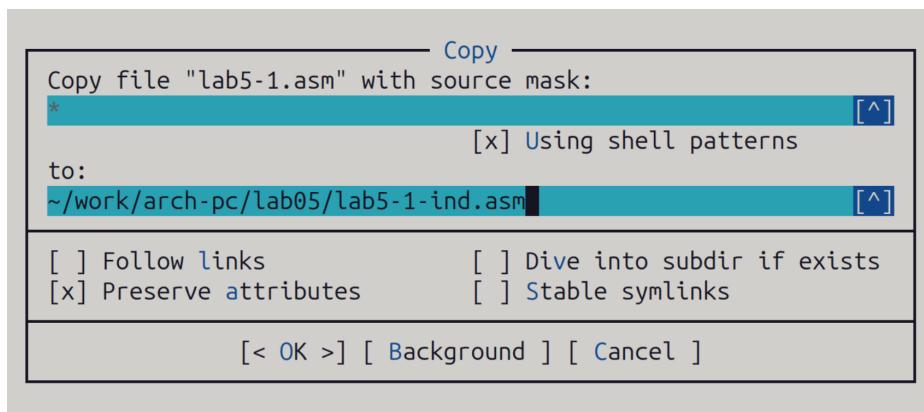


Рисунок 1: Создание копии lab5-1.asm

Наличие файла lab5-1-ind.asm в каталоге подтверждает успешное создание копии исходного файла.

—
Внесение изменений в программу lab5-1-ind.asm без использования внешнего файла in_out.asm в соответствии с заданным алгоритмом.

Листинг программы lab5-1-ind.asm

“asm SECTION .data msg: DB „Введите строку:“, 10 msgLen: EQU \$-msg

```
SECTION .bss buf1: RESB 80
SECTION .text GLOBAL _start
_start: mov eax, 4 mov ebx, 1 mov ecx, msg mov edx, msgLen int 80h

mov eax, 3
mov ebx, 0
mov ecx, buf1
mov edx, 80
int 80h

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

mov eax, 1
mov ebx, 0
int 80h
```

На основе lab5-1.asm после чтения строки с клавиатуры добавлены команды вывода введённой строки: `mov edx, eax` `mov eax, 4` `mov ebx, 1` `mov ecx, buf1` `int 80h` после системного вызова `read` добавлен системный вызов `write`, чтобы вывести на экран строку, только что считанную с клавиатуры.

Получение исполняемого файла из lab5-1-ind.asm и проверка его работы.(рис. Рисунок 2).

```
clanj1@clanj1:~/work/arch-pc/lab05$ nasm -f elf lab5-1-ind.asm -o lab5-1-ind.o  
clanj1@clanj1:~/work/arch-pc/lab05$ ld -m elf_i386 -o lab5-1-ind lab5-1-ind.o  
clanj1@clanj1:~/work/arch-pc/lab05$ ./lab5-1-ind  
Введите строку:  
Лань  
Лань
```

Рисунок 2: Трансляция, компоновка и запуск программы lab5-1-ind

После запуска программа выводит приглашение «Введите строку:», затем принимает введённую строку и выводит её на экран .

—
Создание копии файла lab5-2.asm с новым именем lab5-2-ind.asm.(рис. Рисунок 3).

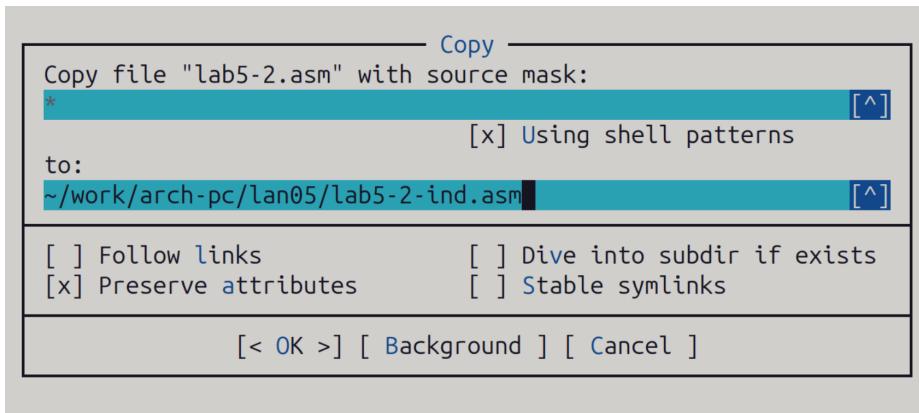


Рисунок 3: Копирование файла lab5-2.asm

В результате в каталоге создан новый файл lab5-2-ind.asm, который является копией исходного файла и может использоваться для дальнейших изменений.

Листинг программы lab5-2-ind.asm

```
“asm %include „in_out.asm“  
SECTION .data msg: DB „Введите строку:“, 0h  
SECTION .bss buf1: RESB 80
```

```

SECTION .text GLOBAL _start
_start: mov eax, msg call sprint

    mov ecx, buf1
    mov edx, 80
    call sread

    mov eax, buf1
    call sprint

    call quit

```

На основе lab5-2.asm после чтения строки с клавиатуры добавлены команды вывода введённой строки: `mov eax, buf1 call sprint`. После чтения строки с клавиатуры с помощью `sread` содержимое буфера `buf1` выводится на экран вызовом подпрограммы `sprint`.

—
Получение исполняемого файла из lab5-2-ind.asm и проверка его работы.(рис. Рисунок 4).

```

clanj1@clanj1:~/work/arch-pc/lab05$ nasm -f elf lab5-2-ind.asm -o lab5-2-ind.o
clanj1@clanj1:~/work/arch-pc/lab05$ ld -m elf_i386 -o lab5-2-ind lab5-2-ind.o
clanj1@clanj1:~/work/arch-pc/lab05$ ./lab5-2-ind
Введите строку:
Лань
Лань

```

Рисунок 4: Трансляция, компоновка и запуск программы lab5-2-ind

На приглашение «Введите строку:» введена фамилия «Лань», программа корректно вывела введённую строку — алгоритм «приглашение → ввод → вывод» выполнен.

Выводы

В ходе выполнения лабораторной работы была достигнута поставленная цель. Были освоены основы работы с файловым менеджером Midnight Commander, а также принципы ввода и вывода данных в программах на языке ассемблера NASM с использованием системных вызовов и подпрограмм.