

ЛО5_Логинова

Архитектура компьютеров

Логинова Дарья Алексеевна

Содержание

Цель работы	1
Задание	1
Теоретическое введение	1
Выполнение лабораторной работы	2
Выводы.....	11
Список литературы.....	11

Цель работы

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

Задание

1. Создайте копию файла `lab5-1.asm`. Внесите изменения в программу (без использования внешнего файла `in_out.asm`), так чтобы она работала по следующему алгоритму: • вывести приглашение типа “Введите строку:”; • ввести строку с клавиатуры; • вывести введенную строку на экран.
2. Получите исполняемый файл и проверьте его работу. На приглашение ввести строку введите свою фамилию.
3. Создайте копию файла `lab5-2.asm`. Исправьте текст программы с использованием подпрограмм из внешнего файла `in_out.asm`, так чтобы она работала по следующему алгоритму: • вывести приглашение типа “Введите строку:”; • ввести строку с клавиатуры; • вывести введенную строку на экран. Не забудьте, подключаемый файл `in_out.asm` должен лежать в том же каталоге, что и файл с программой, в которой он используется.
4. Создайте исполняемый файл и проверьте его работу

Теоретическое введение

Здесь описываются теоретические аспекты, связанные с выполнением работы.

Например, в табл. [-@tbl:std-dir] приведено краткое описание стандартных каталогов Unix.

Описание некоторых каталогов файловой системы GNU Linux {#tbl:std-dir}

Имя каталога	Описание каталога
/	Корневая директория, содержащая всю файловую
/bin	Основные системные утилиты, необходимые как в однопользовательском режиме, так и при обычной работе всем пользователям
/etc	Общесистемные конфигурационные файлы и файлы конфигурации установленных программ
/home	Содержит домашние директории пользователей, которые, в свою очередь, содержат персональные настройки и данные пользователя
/media	Точки монтирования для сменных носителей
/root	Домашняя директория пользователя root
/tmp	Временные файлы
/usr	Вторичная иерархия для данных пользователя

Более подробно про Unix см. в [[@tanenbaum_book_modern-os_ru](#); [@robbins_book_bash_en](#); [@zarrelli_book_mastering-bash_en](#); [@newham_book_learning-bash_en](#)].

Выполнение лабораторной работы

Откроем Midnight Commander и перейдем в созданный каталог. [-@fig:001]

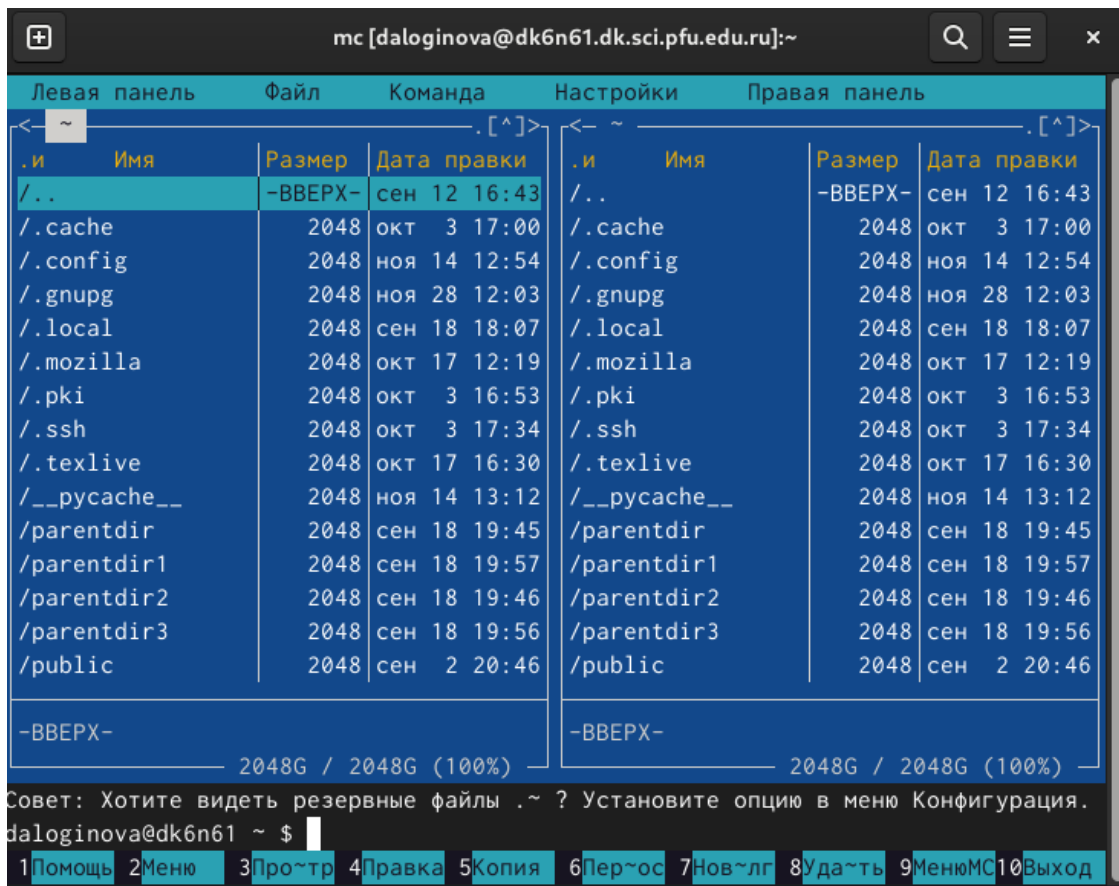


Рис.1

Создадим папку lab05, перейдем в созданный каталог и создадим файл lab5-1.asm. [-@fig:002]

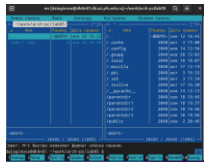
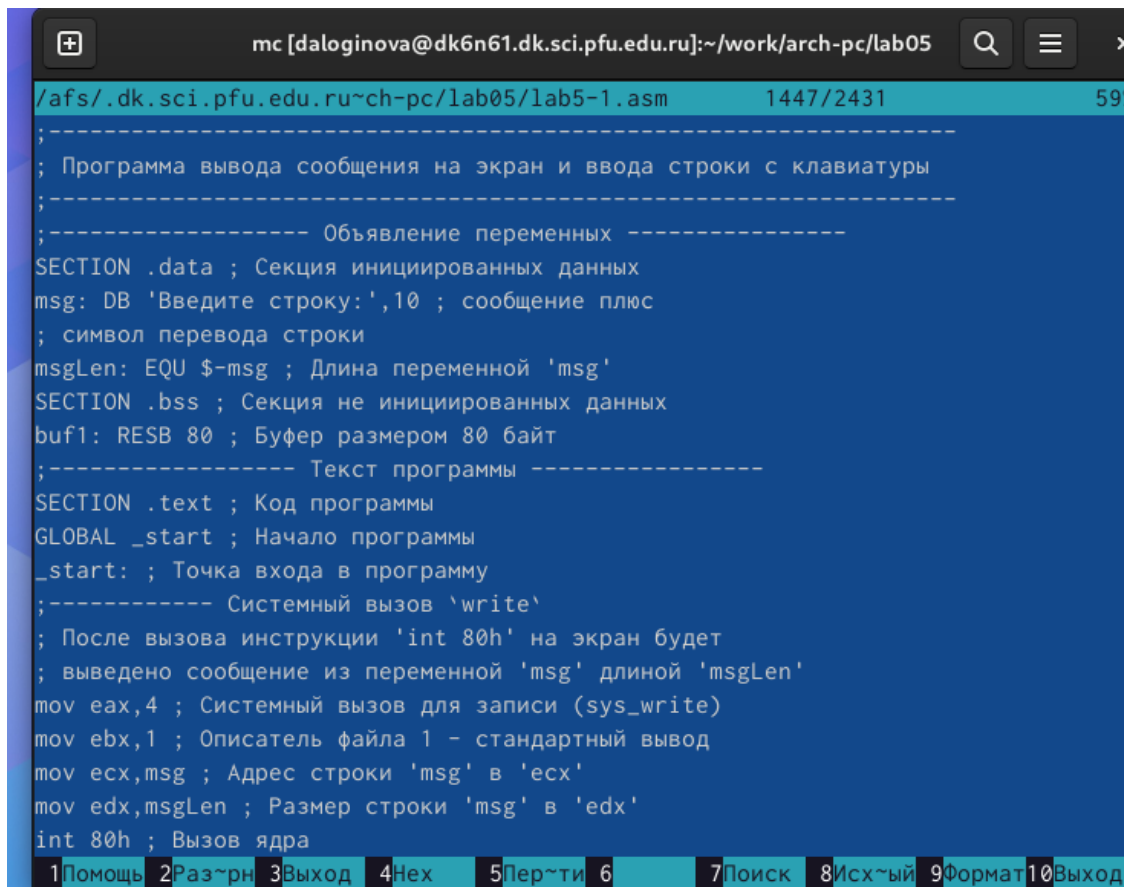


Рис.2

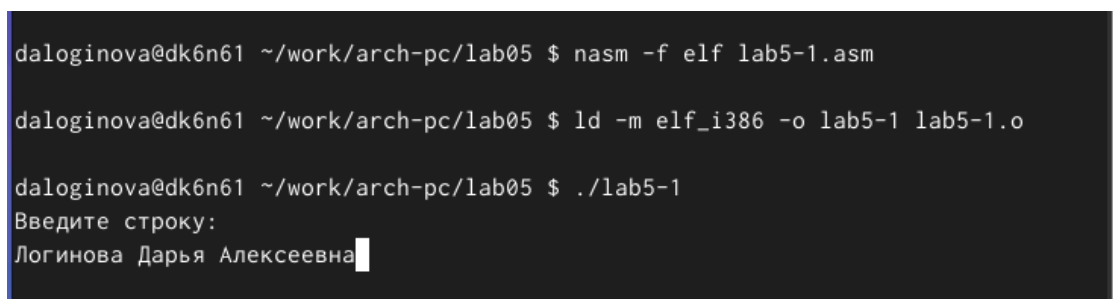
Откроем файл lab5-1.asm и введем текст программы из листинга 5.1. и убедимся, что файл содержит текст программы [-@fig:003]



```
mc [daloginova@dk6n61.dk.sci.pfu.edu.ru]:~/work/arch-pc/lab05
/afs/.dk.sci.pfu.edu.ru~ch-pc/lab05/lab5-1.asm 1447/2431 59
;
; Программа вывода сообщения на экран и ввода строки с клавиатуры
;
;----- Объявление переменных -----
SECTION .data ; Секция инициированных данных
msg: DB 'Введите строку:',10 ; сообщение плюс
; символ перевода строки
msgLen: EQU $-msg ; Длина переменной 'msg'
SECTION .bss ; Секция не инициированных 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 2Разрыв 3Выход 4Hex 5Перти 6 7Поиск 8Исхый 9Формат10Выход
```

Рис.3

Оттранслируем текст программы lab5-1.asm в объектный файл. Выполним компоновку объектного файла и запустим получившийся исполняемый файл. [-@fig:004]



```
daloginova@dk6n61 ~/work/arch-pc/lab05 $ nasm -f elf lab5-1.asm
daloginova@dk6n61 ~/work/arch-pc/lab05 $ ld -m elf_i386 -o lab5-1 lab5-1.o
daloginova@dk6n61 ~/work/arch-pc/lab05 $ ./lab5-1
Введите строку:
Логинова Дарья Алексеевна
```

Рис.4

С помощью функциональной клавиши F6 создадим копию файла lab5-1.asm с именем lab5-2.asm.[-@fig:005]

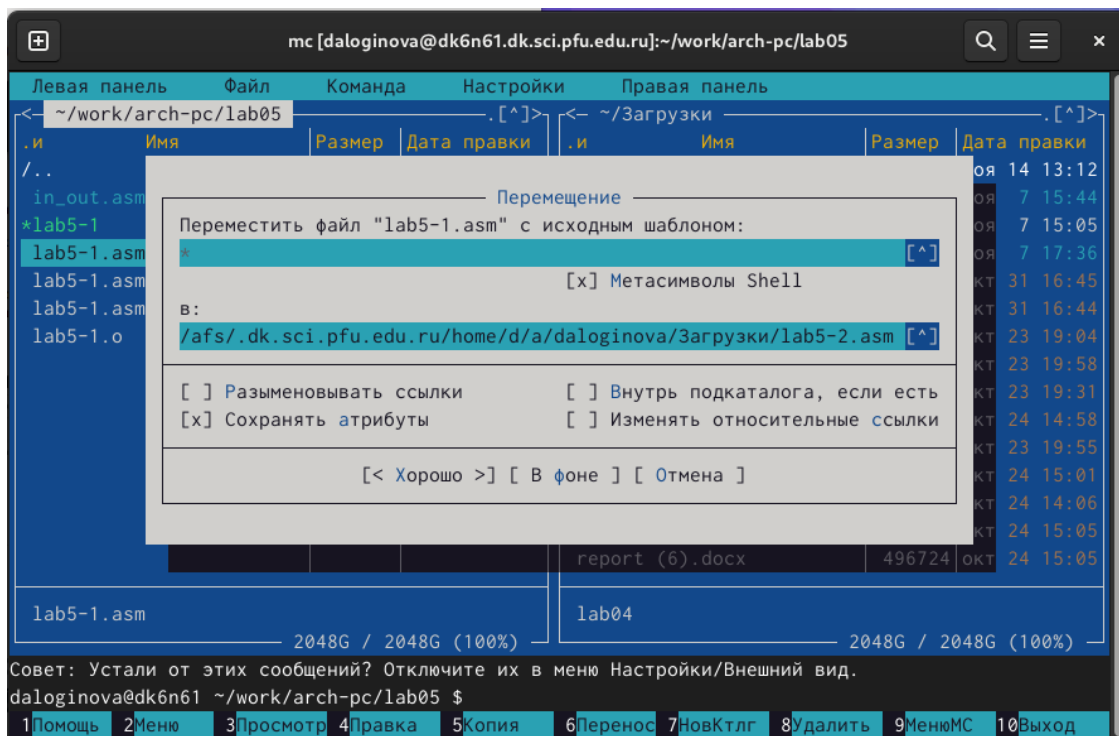


Рис.5

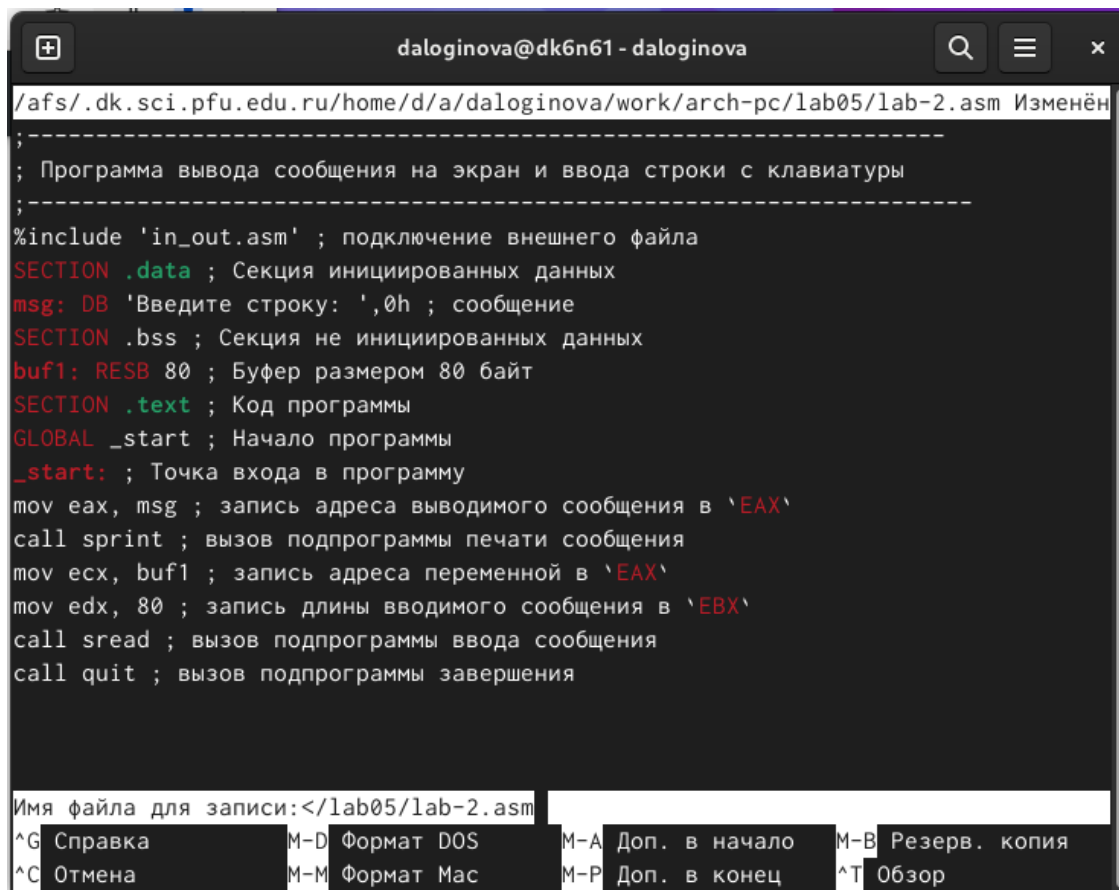
Исправим текст программы в файле lab5-2.asm с использованием подпрограмм из внешнего файла in_out.asm в соответствии с листингом 5.2. Создадим исполняемый файл и проверим его работу. [-@fig:006]

```
daloginova@dk6n61 - daloginova
/afs/.dk.sci.pfu.edu.ru/home/d/a/daloginova/work/arch-pc/lab05/lab-2.asm Изменён
;-----
; Программа вывода сообщения на экран и ввода строки с клавиатуры
;-----
#include 'in_out.asm' ; подключение внешнего файла
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 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

[ Прочитано 17 строк ]
^G Справка    ^O Записать   ^F Поиск      ^K Вырезать   ^T Выполнить  M-U Отмена
^X Выход      ^R ЧитФайл   ^\ Замена     ^U Вставить   ^C Позиция    M-E Повтор
```

Рис.6

В файле lab5-2.asm замените подпрограмму sprintf на sprint. [-@fig:007]



```
dalloginova@dk6n61 - dalloginova
/afs/.dk.sci.pfu.edu.ru/home/d/a/dalloginova/work/arch-pc/lab05/lab-2.asm Изменён
;-----
; Программа вывода сообщения на экран и ввода строки с клавиатуры
;-----
#include 'in_out.asm' ; подключение внешнего файла
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 ; вызов подпрограммы завершения

Имя файла для записи:</lab05/lab-2.asm
^G Справка      M-D Формат DOS   M-A Доп. в начало M-B Резерв. копия
^C Отмена       M-M Формат Mac   M-P Доп. в конец ^T Обзор
```

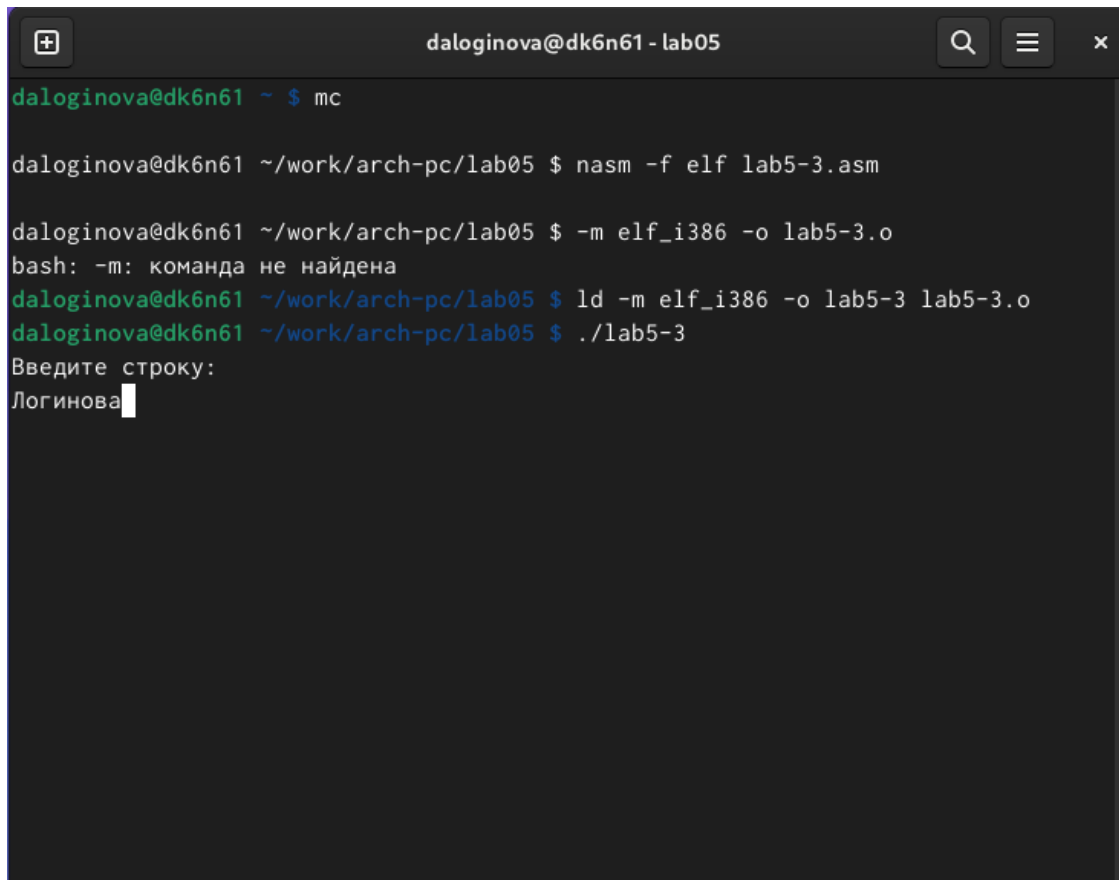
Рис.7

Заметим, что строки вводятся по разному. [-@fig:008]



```
dalloginova@dk6n61 - lab05
dalloginova@dk6n61 ~ $ mc
dalloginova@dk6n61 ~/work/arch-pc/lab05 $ nasm -f elf lab-2.asm
dalloginova@dk6n61 ~/work/arch-pc/lab05 $ ld -m elf_i386 -o lab-2 lab-2.o
dalloginova@dk6n61 ~/work/arch-pc/lab05 $ ./lab-2
Введите строку: 2689
```

@fig:009]

A terminal window titled 'dalloginova@dk6n61 - lab05' with search, menu, and close icons in the title bar. The terminal shows a sequence of commands: 'mc', 'nasm -f elf lab5-3.asm', '-m elf_i386 -o lab5-3.o' (which results in a 'bash: -m: команда не найдена' error), 'ld -m elf_i386 -o lab5-3 lab5-3.o', and './lab5-3'. The prompt 'Введите строку:' is followed by the name 'Логинава' and a cursor.

```
dalloginova@dk6n61 ~ $ mc
dalloginova@dk6n61 ~/work/arch-pc/lab05 $ nasm -f elf lab5-3.asm
dalloginova@dk6n61 ~/work/arch-pc/lab05 $ -m elf_i386 -o lab5-3.o
bash: -m: команда не найдена
dalloginova@dk6n61 ~/work/arch-pc/lab05 $ ld -m elf_i386 -o lab5-3 lab5-3.o
dalloginova@dk6n61 ~/work/arch-pc/lab05 $ ./lab5-3
Введите строку:
Логинава
```

Рис.9

Создадим копию файла lab5-1.asm. Внесите изменения в программу (без использования внешнего файла in_out.asm), так чтобы она работала по алгоритму.Получим исполняемый файл и проверим его работу.[-@fig:010]


```
daloginova@dk6n61 - lab05
daloginova@dk6n61 ~ $ mc
daloginova@dk6n61 ~/work/arch-pc/lab05 $ nasm -f elf lab5-3.asm
daloginova@dk6n61 ~/work/arch-pc/lab05 $ -m elf_i386 -o lab5-3.o
bash: -m: команда не найдена
daloginova@dk6n61 ~/work/arch-pc/lab05 $ ld -m elf_i386 -o lab5-3 lab5-3.o
daloginova@dk6n61 ~/work/arch-pc/lab05 $ ./lab5-3
Введите строку:
логинова
логинова
daloginova@dk6n61 ~/work/arch-pc/lab05 $
```

@fig:011]

[-

```

/afs/.dk.sci.pfu.edu.ru/home/d/a/daloginova/work/arch-pc/lab05/lab5-3.asm
;-----
; Программа вывода сообщения на экран и ввода строки с клавиатуры
;-----
#include 'in_out.asm' ; подключение внешнего файла
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 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
mov eax, buf1 ;
mov ebx, 80 ;
call sprintf
call quit ; вызов подпрограммы завершения

```

Рис.11

Создайте копию файла lab5-2.asm. Исправьте текст программы с использованием подпрограмм из внешнего файла in_out.asm, так чтобы она работала по алгоритму. И проверим его работу. [-@fig:012]

```

daloginova@dk6n61 ~/work/arch-pc/lab05 $ nasm -f elf lab-4.asm
daloginova@dk6n61 ~/work/arch-pc/lab05 $ ld -m elf_i386 -o lab-4 lab-4.o
daloginova@dk6n61 ~/work/arch-pc/lab05 $ ./lab-4
Введите строку: Логина
Логина
daloginova@dk6n61 ~/work/arch-pc/lab05 $

```

@fig:013]

[-

```

;-----
; 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 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 ; вызов подпрограммы завершения

```

Рис.13

Выводы

В ходе работы я приобрела практические навыки работы в Midnight Commander. Освоение инструкций языка ассемблера mov и int.

Список литературы