

Отчет по лабораторной работа №5

Дисциплина: архитектура компьютера

Ширинкин Т. Б.

Содержание

Цель работы	1
Выполнение лабораторной работы	1
Выполнение заданий для самостоятельной работы.....	4

Цель работы

Приобретение навыков работы в Midnight Commander и программирования на ассемблере. Освоение основных инструкций языка ассемблер.

Выполнение лабораторной работы

Открыл Midnight Commander (Рис. [-@fig:000])



Рис. 0 Открыл Midnight Commander

Перешёл в ~/work/arch-pc и создаю папку lab05 (Рис. [-@fig:001])

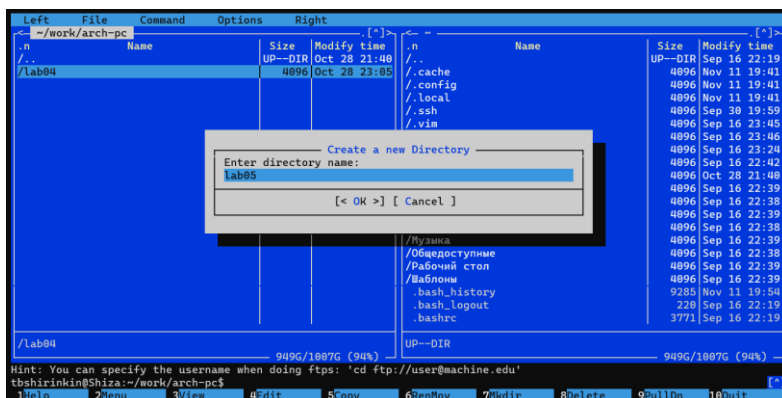


Рис. 1 Перешёл

Создаём файл для программы (Рис. [-@fig:002])

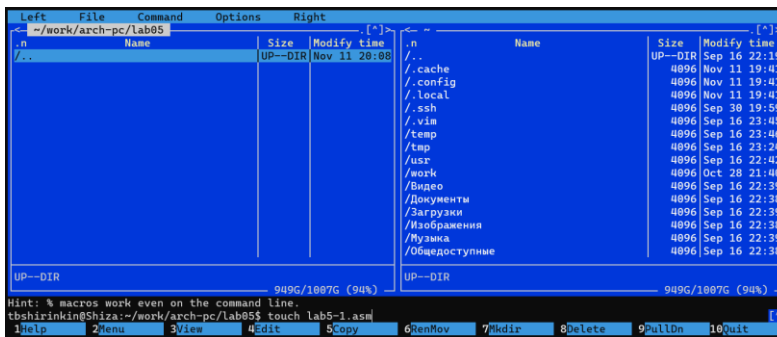


Рис. 2 Создаём

Начинаем писать код (Рис. [-@fig:003])

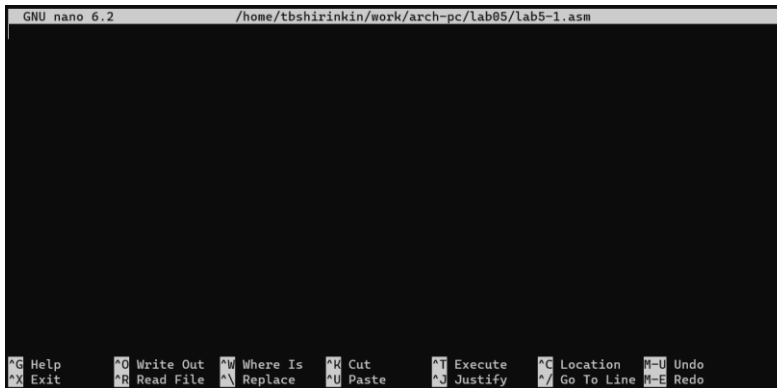


Рис. 3 Начинаем писать

Ввели код (Рис. [-@fig:004])

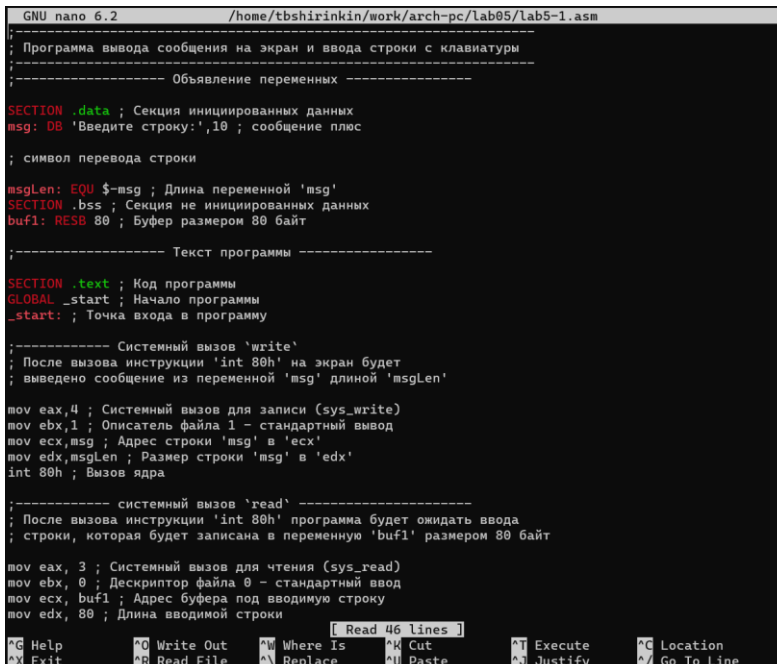


Рис. 4 Ввели

Проверили его введенность (Рис. [-@fig:005])

```
/home/tbshirinkin/work/arch-pc/lab05/lab5-1.asm 2081/2443 85%
; 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
;----- 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 ; Дескриптор файла 0 - стандартный ввод
mov ecx,buf1 ; Адрес буфера под вводимую строку
mov edx,80 ; Длина вводимой строки
int 80h ; Вызов ядра
1Help 2UnWrap 3Quit 4Hex 5Goto 6 7Search 8Raw 9Format 10Quit
```

Рис. 5 Проверили его введенность

Делаем файл объекта (Рис. [-@fig:006])

```
tbshirinkin@Shiza:~/work/arch-pc/lab05$ nasm -f elf lab5-1.asm
```

Рис. 6 Делаем файл объекта

Превращаем в исполняемый (Рис. [-@fig:007])

```
tbshirinkin@Shiza:~/work/arch-pc/lab05$ ld -m elf_i386 -o lab5-1 lab5-1.o
```

Рис. 7 Превращаем в исполняемый

Запускаем программу (Рис. [-@fig:008])

```
tbshirinkin@Shiza:~/work/arch-pc/lab05$ ./lab5-1
```

Рис. 8 Запускаем программу

Проверил, вводится (Рис. [-@fig:009])

```
Введите строку:
Ширинкин Тимур Борисович
```

Рис. 9 Проверил, вводится

Скачал и переместил файл in_out.asm, скопировал lab5-1 в lab5-2, скопировал новый код и вставил в lab5-2 (Рис. [-@fig:010])

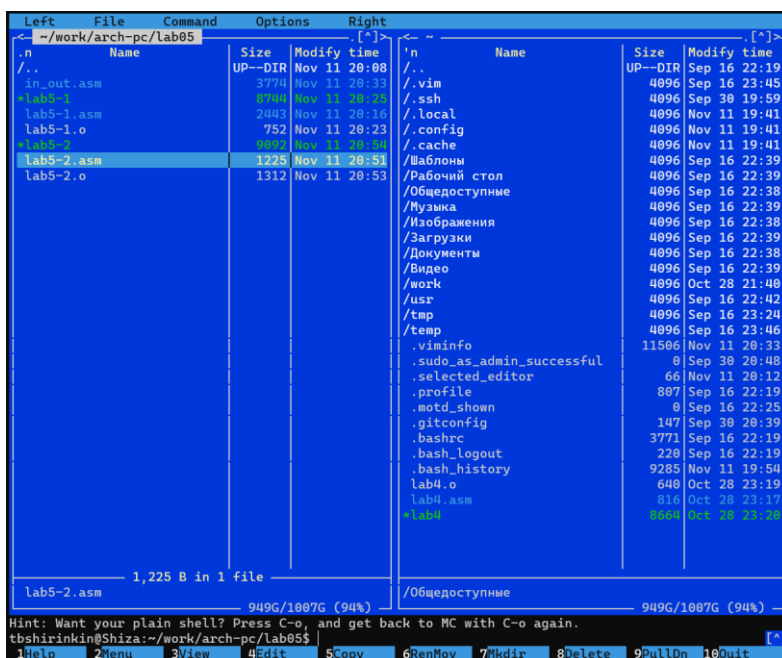


Рис. 10 Скачал и переместил файл `in_out.asm`, скопировал `lab5-1` в `lab5-2`, скопировал новый код и вставил в `lab5-2`

Собираем и проверяем (Рис. [-@fig:011])

```
tbshirinkin@Shiza:~/work/arch-pc/lab05$ nasm -f elf lab5-2.asm
tbshirinkin@Shiza:~/work/arch-pc/lab05$ ld -m elf_i386 lab5-2.o -o lab5-2
tbshirinkin@Shiza:~/work/arch-pc/lab05$ ./lab5-2
Введите строку:
Ширинкин Тимур Борисович
tbshirinkin@Shiza:~/work/arch-pc/lab05$
```

Рис. 11 Собираем и проверяем

Выполнение заданий для самостоятельной работы

Копирую `lab5-1` в `lab5-3` и редактирую файл `lab5-3`: после блока ввода текста, вставил блок вывода текста, но из буфера с введённым текстом (Рис. [-@fig:012])

```
GNU nano 6.2 /home/tbshirinkin/work/arch-pc/lab05/lab5-3.asm
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
;----- 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 ; Дескриптор файла 0 - стандартный ввод
mov ecx,buf1 ; Адрес буфера под вводимую строку
mov edx,80 ; Длина вводимой строки
int 80h ; Вызов ядра
;----- Системный вызов 'write'
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 ; lool
;
^G Help ^O Write Out ^W Where Is ^K Cut ^T Execute ^C Location ^M-U Undo
^X Exit ^R Read File ^\ Replace ^U Paste ^J Justify ^/_ Go To Line ^M-E Redo
```

Рис. 12 Копирую lab5-1 в lab5-3 и редактирую файл lab5-3: после блока ввода текста, вставил блок вывода текста, но из буфера с введённым текстом

Собираю и проверяю lib5-3: Ввёл ФИО и оно повторно высветилось (Рис. [-@fig:013])

```
tbshirinkin@Shiza:~/work/arch-pc/lab05$ nasm -f elf lab5-3.asm
tbshirinkin@Shiza:~/work/arch-pc/lab05$ ld -m elf_i386 lab5-3.o -o lab5-3
tbshirinkin@Shiza:~/work/arch-pc/lab05$ ./lab5-3
Введите строку:
Ширинкин Тимур Борисович
Ширинкин Тимур Борисович
tbshirinkin@Shiza:~/work/arch-pc/lab05$ |
```

Рис. 13 Собираю и проверяю lib5-3: Ввёл ФИО и оно повторно высветилось

Аналогично делаем с lib5-2: копируем и изменяем (Рис. [-@fig:014])

```
GNU nano 6.2 /home/tbshirinkin/work/arch-pc/lab05/lab5-4.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:',10 ; 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 выводимого сообщения в 'EBX'
call sread ; вызов подпрограммы ввода сообщения
mov eax,buf1 ; запись адреса выводимого сообщения в 'EAX'
call sprintf ; вызов подпрограммы печати сообщения
call quit ; вызов подпрограммы завершения
;
[ Read 19 lines ]
^G Help ^O Write Out ^W Where Is ^K Cut ^T Execute ^C Location ^M-U Undo
^X Exit ^R Read File ^\ Replace ^U Paste ^J Justify ^/_ Go To Line
```

Рис. 14 Аналогично делаем с lib5-2: копируем и изменяем

Собрал, проверил - работает. (Рис. [-@fig:015])

```
nasm -f elf lab5-4.asm
tbshirinkin@Shiza:~/work/arch-pc/lab05$ ld -m elf_i386 lab5-4.o -o lab5-4
tbshirinkin@Shiza:~/work/arch-pc/lab05$ ./lab5-4
Введите строку:
Ширинкин Тимур Борисович
Ширинкин Тимур Борисович
tbshirinkin@Shiza:~/work/arch-pc/lab05$ |
```

Выводы

Цель достигнута: приобретены навыки работы в Midnight Commander и программирования на ассемблере, освоены основные инструкции языка ассемблер.