

# **Отчёт по лабораторной работе 5**

**Архитектура компьютеров**

Нурбердиев Гурбанмырад НКАбд-03-24

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Выполнение лабораторной работы</b>	<b>6</b>
2.1	Знакомство с Midnight Commander . . . . .	6
2.2	Подключение внешнего файла in_out.asm . . . . .	11
2.3	Задание для самостоятельной работы . . . . .	14
<b>3</b>	<b>Выводы</b>	<b>17</b>

## Список иллюстраций

2.1	Запуск Midnight Commander . . . . .	6
2.2	Создание каталога . . . . .	7
2.3	Создание файла lab05-1.asm . . . . .	7
2.4	выбираю редактор . . . . .	8
2.5	Программа lab05-1.asm . . . . .	9
2.6	Просмотр файла lab05-1.asm . . . . .	10
2.7	Запуск программы lab05-1.asm . . . . .	10
2.8	Копирование файла in_out.asm . . . . .	11
2.9	Копирование файла lab05-1.asm . . . . .	11
2.10	Программа lab05-2.asm . . . . .	12
2.11	Запуск программы lab05-2.asm . . . . .	12
2.12	Программа в файле lab05-2.asm . . . . .	13
2.13	Запуск программы lab05-2.asm . . . . .	13
2.14	Программа lab05-3.asm . . . . .	14
2.15	Запуск программы lab05-3.asm . . . . .	15
2.16	Программа lab05-4.asm . . . . .	16
2.17	Запуск программы lab05-4.asm . . . . .	16

## Список таблиц

# 1 Цель работы

Целью работы является приобретение практических навыков работы в Midnight Commander. Освоение инструкций языка ассемблера `mov` и `int`.

## 2 Выполнение лабораторной работы

### 2.1 Знакомство с Midnight Commander

Запускаю Midnight Commander (см. рис. 2.1), используя клавиши со стрелками и Enter, перехожу в каталог ~/work/arch-рс. Затем нажимаю F7 для создания нового каталога под названием lab05 (см. рис. 2.2).

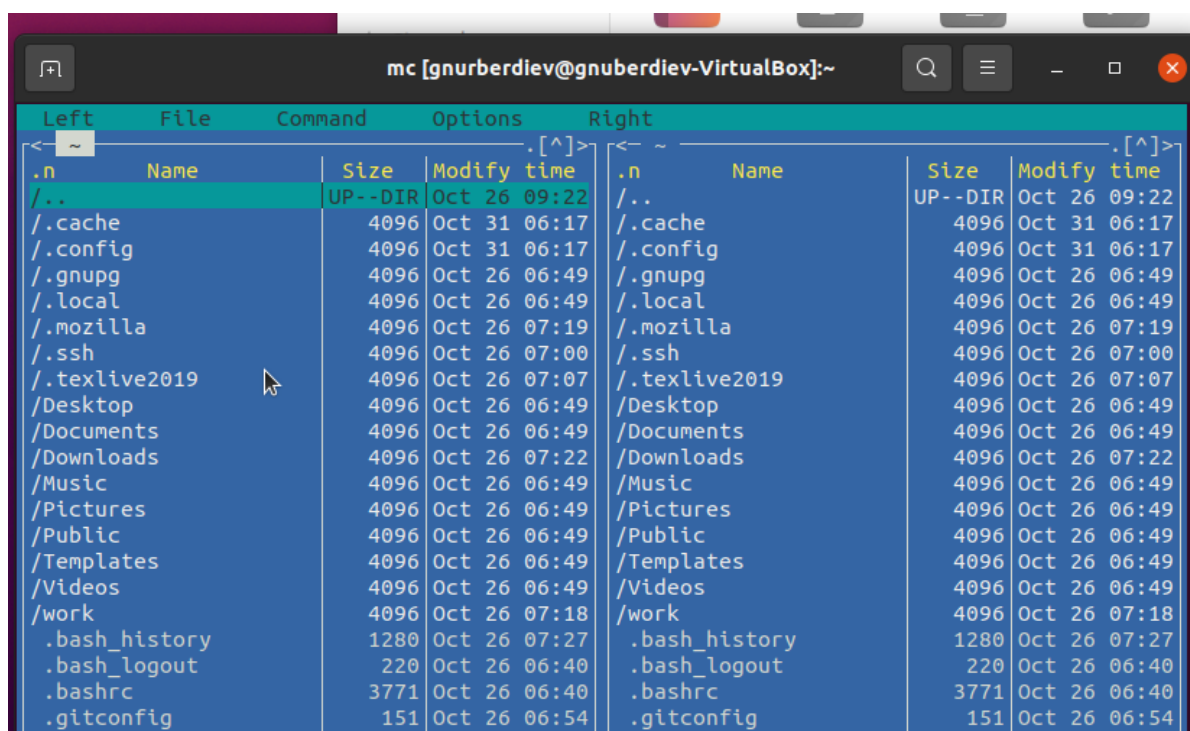


Рис. 2.1: Запуск Midnight Commander

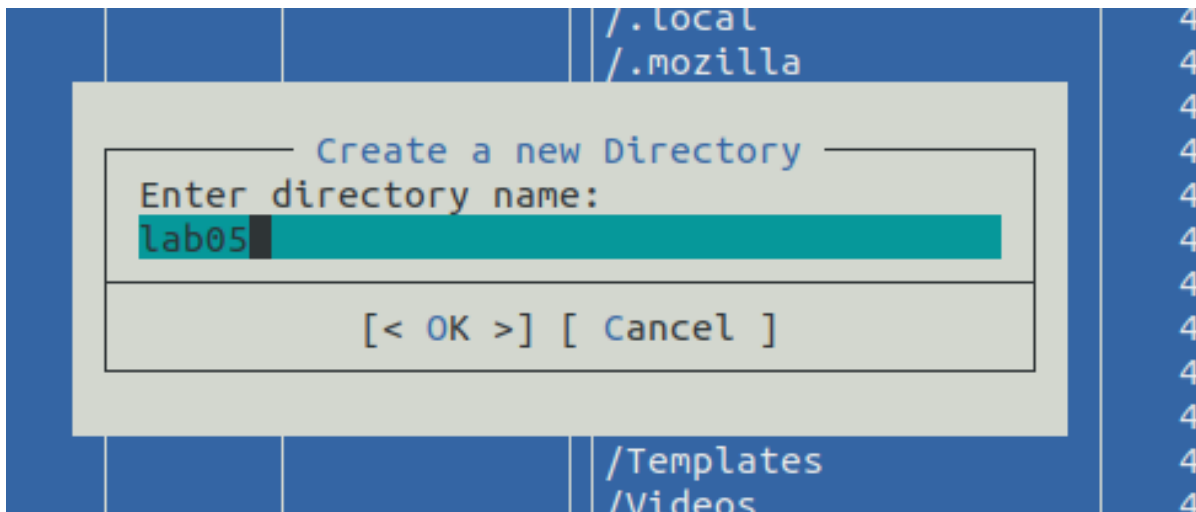


Рис. 2.2: Создание каталога

С помощью команды touch создаю файл lab05-1.asm (см. рис. 2.3).

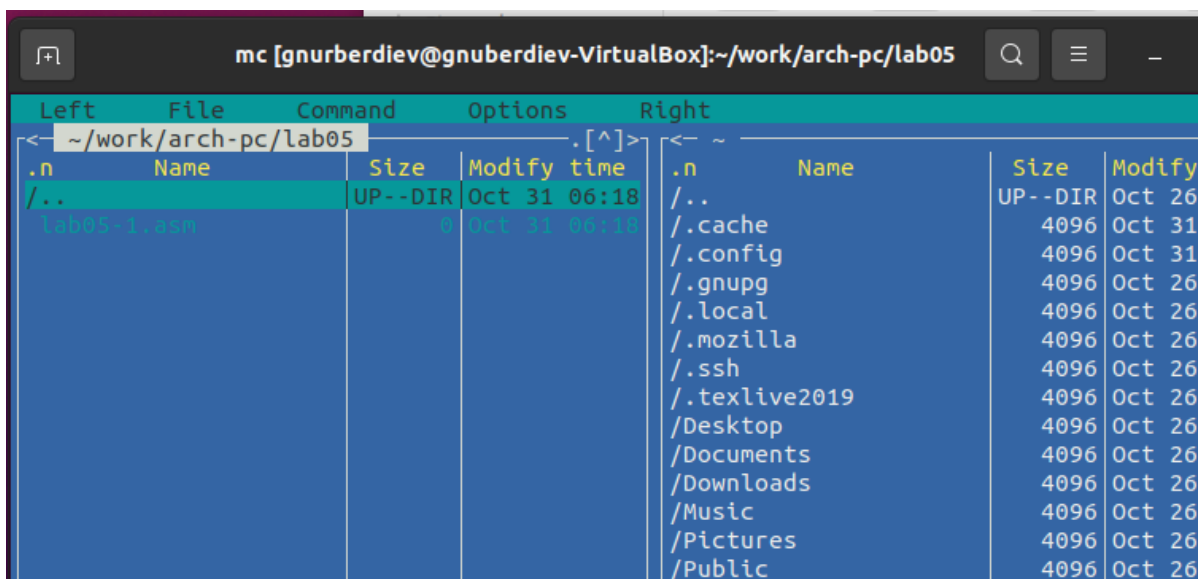


Рис. 2.3: Создание файла lab05-1.asm

Открываю файл для редактирования, нажав F4, и выбираю редактор mcedit, после чего пишу код программы согласно заданию (см. рис. 2.5).

```
gnurberdiev@gnurberdiev-VirtualBox:~$ mc
gnurberdiev@gnurberdiev-VirtualBox:~/work/arch-pc/lab05$ touch lab05-1.asm
Select an editor. To change later, run 'select-editor'.
 1. /bin/nano          <---- easiest
 2. /usr/bin/mcedit
 3. /usr/bin/vim.tiny
 4. /usr/bin/emacs
 5. /bin/ed
Choose 1-5 [1]: █
```

Рис. 2.4: выбираю редактор



```
/home/gnurber~5/lab05-1.asm [----] 9 L:[ 1+19
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
```

Рис. 2.5: Программа lab05-1.asm

Для проверки содержимого файла открываю его на просмотр, нажав F3, и убеждаюсь, что код написан верно (см. рис. 2.6).

```
/home/gnurberdiev/work/arch-pc/lab05/lab05-1.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
```

Рис. 2.6: Просмотр файла lab05-1.asm

Транслирую файл программы в объектный файл, а затем выполняю компоновку, в результате чего получаю исполняемый файл программы (см. рис. 2.7).

```
gnurberdiev@gnurberdiev-VirtualBox:~/work/arch-pc/lab05$
gnurberdiev@gnurberdiev-VirtualBox:~/work/arch-pc/lab05$ nasm -f elf lab05-1.asm
gnurberdiev@gnurberdiev-VirtualBox:~/work/arch-pc/lab05$ ld -m elf_i386 lab05-1.o -o lab05-1
gnurberdiev@gnurberdiev-VirtualBox:~/work/arch-pc/lab05$ ./lab05-1
Введите строку:
ubuntu
gnurberdiev@gnurberdiev-VirtualBox:~/work/arch-pc/lab05$
```

Рис. 2.7: Запуск программы lab05-1.asm

## 2.2 Подключение внешнего файла in\_out.asm

Скачиваю файл in\_out.asm и размещаю его в рабочем каталоге (см. рис. 2.8). Для копирования файла использую клавишу F5, а для перемещения — клавишу F6.

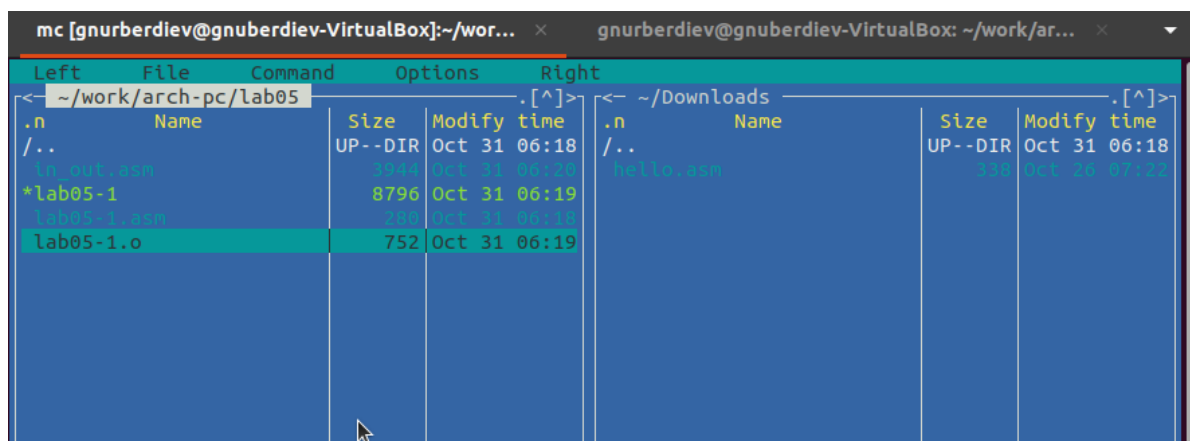


Рис. 2.8: Копирование файла in\_out.asm

Копирую файл lab05-1.asm, создавая его копию под именем lab05-2.asm (см. рис. 2.9).

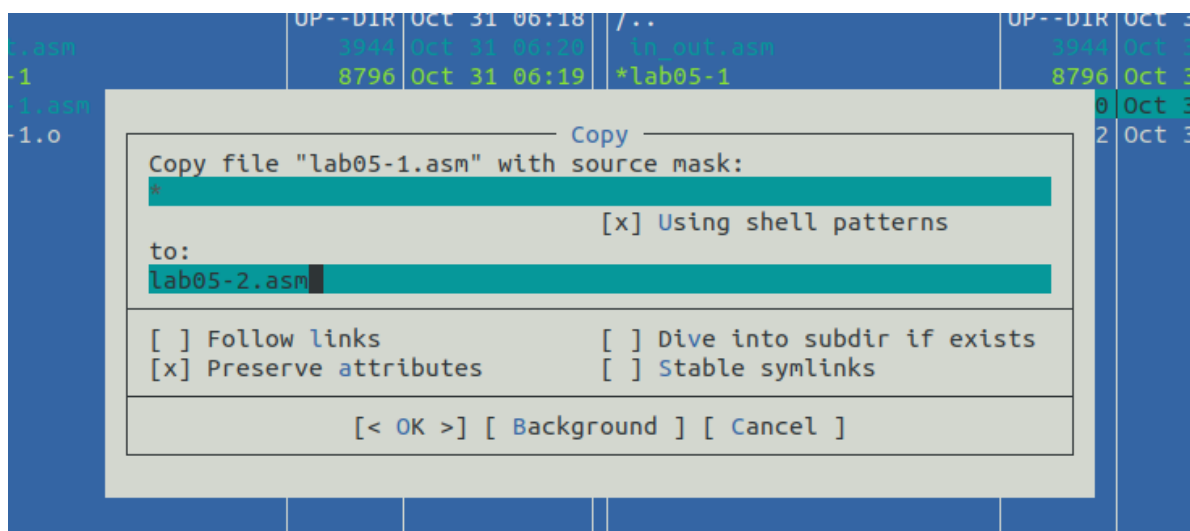


Рис. 2.9: Копирование файла lab05-1.asm

Пишу код для программы lab05-2.asm, используя подпрограммы из внешнего файла in\_out.asm (см. рис. 2.10).

```
/home/gnurberdiev~ab05/lab05-2.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
call quit
```

Рис. 2.10: Программа lab05-2.asm

Компилирую программу и проверяю её запуск (см. рис. 2.11).

```
gnurberdiev@gnurberdiev-VirtualBox:~/work/arch-pc/lab05$
gnurberdiev@gnurberdiev-VirtualBox:~/work/arch-pc/lab05$ nasm -f elf lab05-2.asm
gnurberdiev@gnurberdiev-VirtualBox:~/work/arch-pc/lab05$ ld -m elf_i386 lab05-2.o -o lab05-2
gnurberdiev@gnurberdiev-VirtualBox:~/work/arch-pc/lab05$ ./lab05-2
Введите строку:
ubuntu
gnurberdiev@gnurberdiev-VirtualBox:~/work/arch-pc/lab05$ █
```

Рис. 2.11: Запуск программы lab05-2.asm

В файле lab05-2.asm заменяю подпрограмму sprintLF на sprint. После этого заново собираю исполняемый файл (см. рис. 2.12 и 2.13).

```
/home/gnurberdiev~ab05/lab05-2.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 sprint
mov ecx, buf1
mov edx, 80
call sread
call quit
```

Рис. 2.12: Программа в файле lab05-2.asm

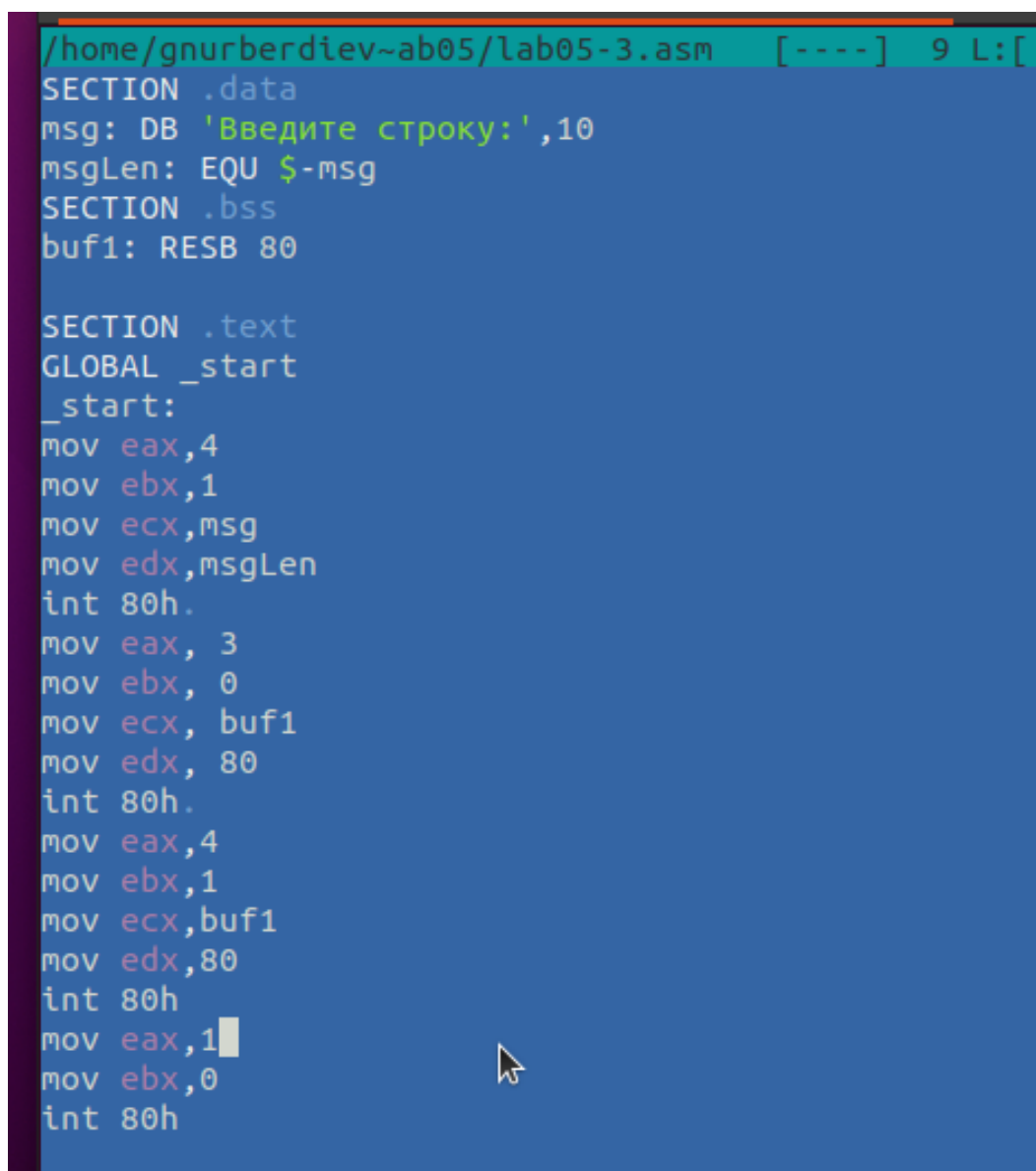
```
gnurberdiev@gnurberdiev-VirtualBox:~/work/arch-pc/lab05$ nasm -f elf lab05-2.asm
gnurberdiev@gnurberdiev-VirtualBox:~/work/arch-pc/lab05$ ld -m elf_i386 lab05-2.o -o lab05-2
gnurberdiev@gnurberdiev-VirtualBox:~/work/arch-pc/lab05$ ./lab05-2
Введите строку: ubuntu
gnurberdiev@gnurberdiev-VirtualBox:~/work/arch-pc/lab05$ █
```

Рис. 2.13: Запуск программы lab05-2.asm

Теперь программа выводит строку без перехода на новую строку в конце.

## 2.3 Задание для самостоятельной работы

Копирую программу lab05-1.asm и модифицирую код, чтобы она работала по следующему алгоритму (см. рис. 2.14 и 2.15): - выводит приглашение “Введите строку:”; - принимает строку с клавиатуры; - отображает введенную строку на экране.



```
/home/gnurberdiev~ab05/lab05-3.asm  [ - - - ]  9 L: [
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,4
mov ebx,1
mov ecx,buf1
mov edx,80
int 80h
mov eax,1
mov ebx,0
int 80h
```

Рис. 2.14: Программа lab05-3.asm

```
gnurberdiev@gnurberdiev-VirtualBox:~/work/arch-pc/lab05$  
gnurberdiev@gnurberdiev-VirtualBox:~/work/arch-pc/lab05$ nasm -f elf lab05-3.asm  
gnurberdiev@gnurberdiev-VirtualBox:~/work/arch-pc/lab05$ ld -m elf_i386 lab05-3.o -o lab05-3  
gnurberdiev@gnurberdiev-VirtualBox:~/work/arch-pc/lab05$ ./lab05-3  
Введите строку:  
ubuntu  
ubuntu  
gnurberdiev@gnurberdiev-VirtualBox:~/work/arch-pc/lab05$ █
```

Рис. 2.15: Запуск программы lab05-3.asm

Аналогично, копирую программу lab05-2.asm и изменяю код, теперь используя подпрограммы из файла in\_out.asm (см. рис. 2.16 и 2.17).

```
/home/gnurberdiev~ab05/lab05-4.asm [----] 9
%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
```

Рис. 2.16: Программа lab05-4.asm

```
gnurberdiev@gnurberdiev-VirtualBox:~/work/arch-pc/lab05$
gnurberdiev@gnurberdiev-VirtualBox:~/work/arch-pc/lab05$ nasm -f elf lab05-4.asm
gnurberdiev@gnurberdiev-VirtualBox:~/work/arch-pc/lab05$ ld -m elf_i386 lab05-4.o -o lab05-4
gnurberdiev@gnurberdiev-VirtualBox:~/work/arch-pc/lab05$ ./lab05-4
Введите строку: ubuntu
gnurberdiev@gnurberdiev-VirtualBox:~/work/arch-pc/lab05$
```

Рис. 2.17: Запуск программы lab05-4.asm



## 3 Выводы

Научились писать базовые ассемблерные программы. Освоили ассемблерные инструкции `mov` и `int`.