

# Отчет по лабораторной работе №6

Архитектура компьютера

Копылова Виктория Валерьевна

## Содержание

Цель работы.....	1
Задание .....	1
Теоретическое введение .....	1
Выполнение лабораторной работы .....	2
Выводы.....	10
Список литературы .....	10

## Цель работы

Освоение арифметических инструкций языка ассемблера NASM.

## Задание

Написать программу вычисления выражения  $\square = \square(\square)$ . Программа должна выводить выражение для вычисления, выводить запрос на ввод значения  $\square$ , вычислять заданное выражение в зависимости от введенного  $\square$ , выводить результат вычислений. Вид функции  $\square(\square)$  выбрать из таблицы 6.3 вариантов заданий в соответствии с номером полученным при выполнении лабораторной работы. Создайте исполняемый файл и проверьте его работу для значений  $\square 1$  и  $\square 2$  из 6.3.

## 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

катало

га

Описание каталога

/

Корневая директория, содержащая всю файловую

/bin

Основные системные утилиты, необходимые как в однопользовательском режиме, так и при обычной работе всем пользователям

Имя каталога	Описание каталога
/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

1. Создайте каталог для программ лабораторной работы № 6, перейдите в него и создайте файл lab6-1.asm: (рис.1 [-@fig:001]).

```
vvkopihlova@dk3n55 ~ $ mkdir ~/work/arch-pc/lab06
vvkopihlova@dk3n55 ~ $ cd ~/work/arch-pc/lab06
vvkopihlova@dk3n55 ~/work/arch-pc/lab06 $ touch lab6-1.asm
vvkopihlova@dk3n55 ~/work/arch-pc/lab06 $
```

рис.1

2. Рассмотрим примеры программ вывода символьных и численных значений. Программы будут выводить значения записанные в регистр eax (рис.2 [-@fig:002]), (рис.3 [-@fig:003]).



```
....dk.sci.pfu.edu.ru/home/v/v/vvkopihlova/work/arch-pc/lab06/lab6-1.asm Изменён
#include 'in_out.asm'
SECTION .bss
buf1: RESB 80
SECTION .text
GLOBAL _start
_start:
mov eax,'6'
mov ebx,'4'
add eax,ebx
mov [buf1],eax
mov eax,buf1
call printf
call _exit
```

рис.2

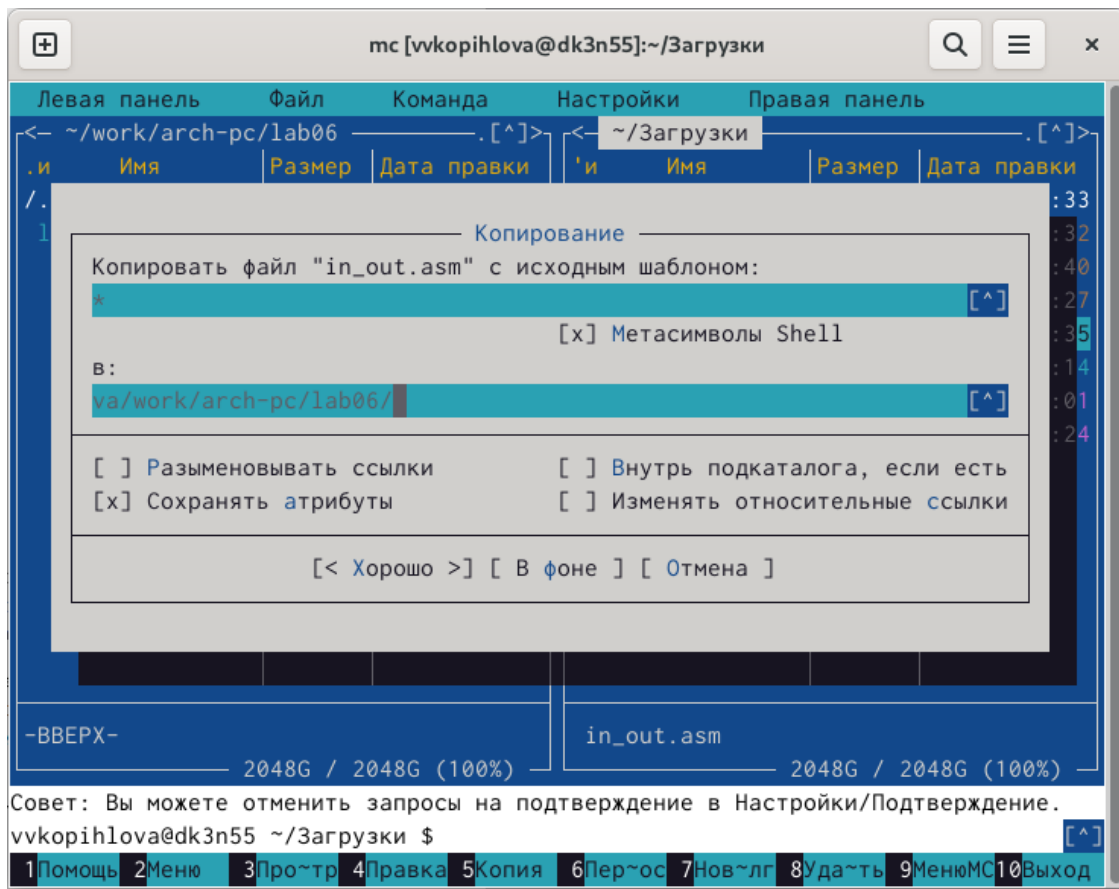


рис.3

Создайте исполняемый файл и запустите его (рис.4 [-@fig:004])

```

vvkopihirova@dk3n55 ~/work/arch-pc/lab06 $ nasm -f elf lab6-1.asm
vvkopihirova@dk3n55 ~/work/arch-pc/lab06 $ ld -m elf_i386 -o lab6-1 lab6-1.o
vvkopihirova@dk3n55 ~/work/arch-pc/lab06 $ ./lab6-1
j
vvkopihirova@dk3n55 ~/work/arch-pc/lab06 $ █

```

рис.4

- Далее изменим текст программы и вместо символов, запишем в регистры числа. Исправьте текст программы (Листинг 6.1) следующим образом (рис.5 [-@fig:005])



```

/afs/.dk.sci.pfu.edu.ru/home/v/v/vvkopihirova/work/arch-pc/lab06/lab6-1.asm
%include 'in_out.asm'
SECTION .bss
buf1: RESB 80
SECTION .text
GLOBAL _start
_start:
mov eax,6
mov ebx,4
add eax,ebx
mov [buf1],eax
mov eax,buf1
call sprintLF
call quit

```

рис.5

Создайте исполняемый файл и запустите его (рис.6 [-@fig:006])

```

vvkopihirova@dk3n55 ~/work/arch-pc/lab06 $ nasm -f elf lab6-1.asm
vvkopihirova@dk3n55 ~/work/arch-pc/lab06 $ ld -m elf_i386 -o lab6-1 lab6-1.o
vvkopihirova@dk3n55 ~/work/arch-pc/lab06 $ ./lab6-1
j

vvkopihirova@dk3n55 ~/work/arch-pc/lab06 $

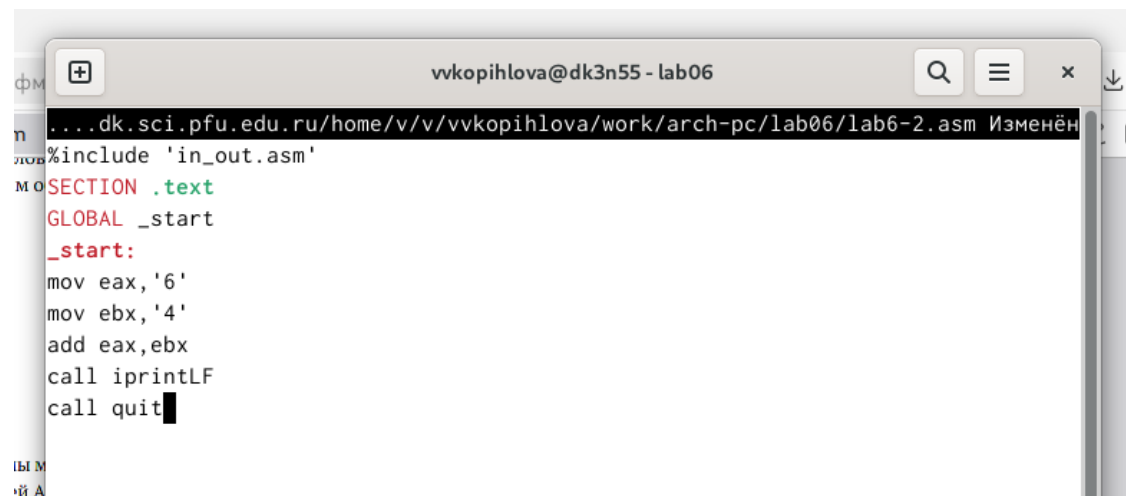
```

рис.6

Создайте файл lab6-2.asm в каталоге ~/work/arch-pc/lab06 и введите в него текст программы из листинга 6.2 (рис.6 [-@fig:006])

```
vkopihlova@dk3n55 ~/work/arch-pc/lab06 $ touch lab6-2.asm
```

рис.7



```
vkopihlova@dk3n55 - lab06
...dk.sci.pfu.edu.ru/home/v/v/vkopihlova/work/arch-pc/lab06/lab6-2.asm Изменён
#include 'in_out.asm'
SECTION .text
GLOBAL _start
_start:
mov eax, '6'
mov ebx, '4'
add eax, ebx
call iprintLF
call quit
```

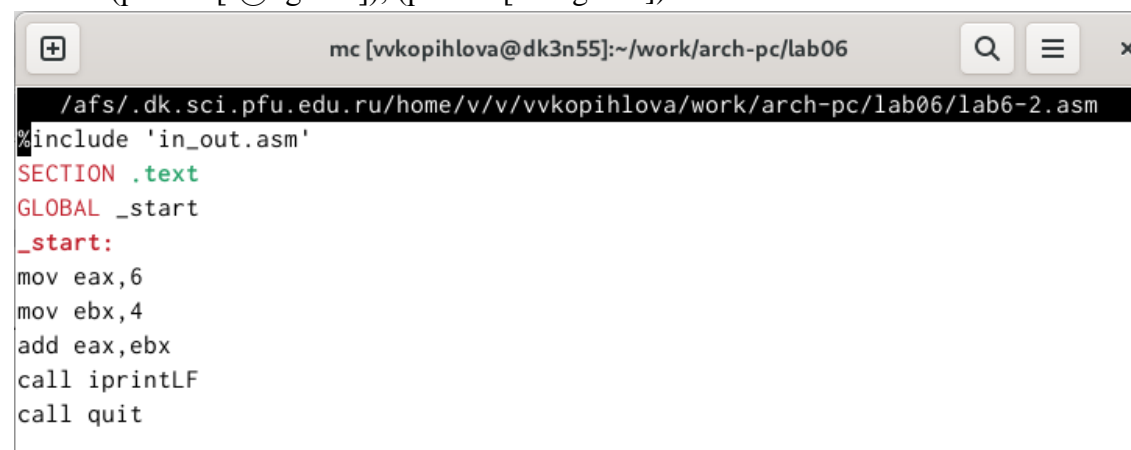
рис.8

Создайте исполняемый файл и запустите его (рис.9 [-@fig:009])

```
vkopihlova@dk3n55 ~/work/arch-pc/lab06 $ nasm -f elf lab6-2.asm
vkopihlova@dk3n55 ~/work/arch-pc/lab06 $ ld -m elf_i386 -o lab6-2 lab6-2.o
vkopihlova@dk3n55 ~/work/arch-pc/lab06 $ ./lab6-2
106
```

рис.9

5. Аналогично предыдущему примеру изменим символы на числа. Замените строки (рис.10 [-@fig:010]), (рис.11 [-@fig:011])



```
mc [vkopihlova@dk3n55]:~/work/arch-pc/lab06
/afs/.dk.sci.pfu.edu.ru/home/v/v/vkopihlova/work/arch-pc/lab06/lab6-2.asm
#include 'in_out.asm'
SECTION .text
GLOBAL _start
_start:
mov eax, 6
mov ebx, 4
add eax, ebx
call iprintLF
call quit
```

рис.10

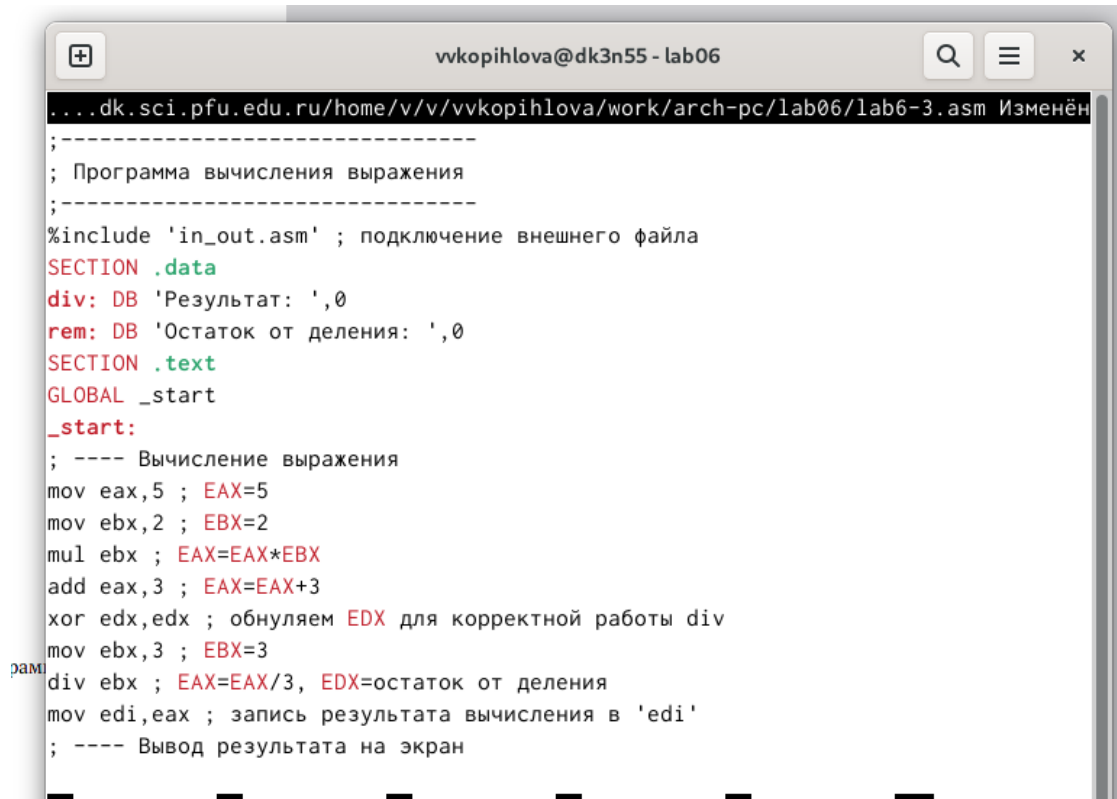
```

vvkopihirova@dk3n55 ~/work/arch-pc/lab06 $ nasm -f elf lab6-2.asm
vvkopihirova@dk3n55 ~/work/arch-pc/lab06 $ ld -m elf_i386 -o lab6-2 lab6-2.o
vvkopihirova@dk3n55 ~/work/arch-pc/lab06 $ ./lab6-2
10

```

рис.11

Создайте файл lab6-3.asm в каталоге ~/work/arch-pc/lab06: (рис.12 [-@fig:012]), (рис.13 [-@fig:013])



```

...dk.sci.pfu.edu.ru/home/v/v/vvkopihirova/work/arch-pc/lab06/lab6-3.asm Изменён
;-----
; Программа вычисления выражения
;-----
%include 'in_out.asm' ; подключение внешнего файла
SECTION .data
div: DB 'Результат: ',0
rem: DB 'Остаток от деления: ',0
SECTION .text
GLOBAL _start
_start:
; ---- Вычисление выражения
mov eax,5 ; EAX=5
mov ebx,2 ; EBX=2
mul ebx ; EAX=EAX*EBX
add eax,3 ; EAX=EAX+3
xor edx,edx ; обнуляем EDX для корректной работы div
mov ebx,3 ; EBX=3
div ebx ; EAX=EAX/3, EDX=остаток от деления
mov edi,eax ; запись результата вычисления в 'edi'
; ---- Вывод результата на экран

```

рис.12

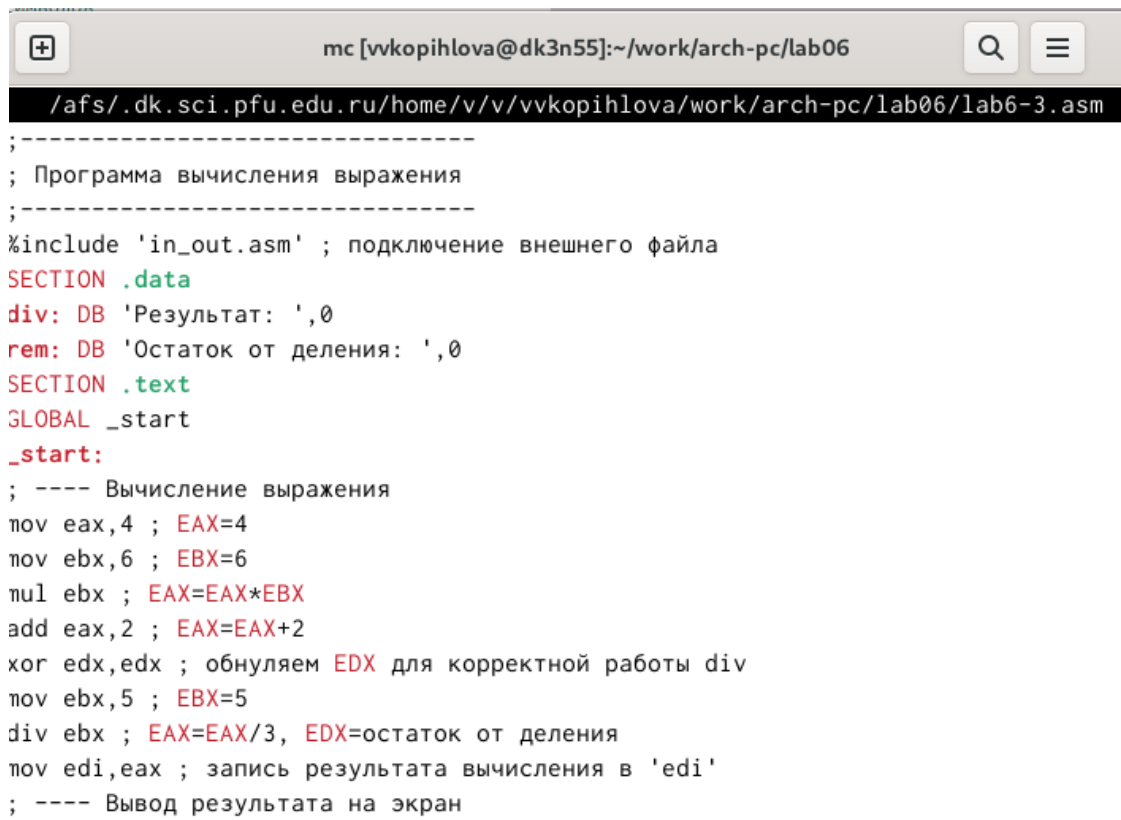
```

vvkopihirova@dk3n55 ~/work/arch-pc/lab06 $ nasm -f elf lab6-3.asm
vvkopihirova@dk3n55 ~/work/arch-pc/lab06 $ ld -m elf_i386 -o lab6-3 lab6-3.o
vvkopihirova@dk3n55 ~/work/arch-pc/lab06 $ ./lab6-3
Результат: 4
Остаток от деления: 1

```

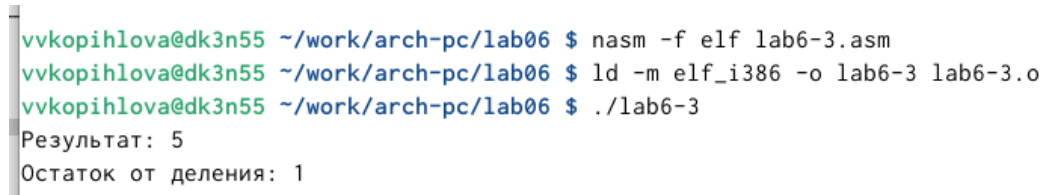
рис.13

Измените текст программы для вычисления выражения  $\square(\square) = (4 * 6 + 2)/5$ . Создайте исполняемый файл и проверьте его работу. (рис.14 [-@fig:014]), (рис.15 [-@fig:015])



```
mc[vvkopihlova@dk3n55]:~/work/arch-pc/lab06
/afs/.dk.sci.pfu.edu.ru/home/v/v/vvkopihlova/work/arch-pc/lab06/lab6-3.asm
;
; Программа вычисления выражения
;
%include 'in_out.asm' ; подключение внешнего файла
SECTION .data
div: DB 'Результат: ',0
rem: DB 'Остаток от деления: ',0
SECTION .text
GLOBAL _start
_start:
; ---- Вычисление выражения
mov eax,4 ; EAX=4
mov ebx,6 ; EBX=6
mul ebx ; EAX=EAX*EBX
add eax,2 ; EAX=EAX+2
xor edx,edx ; обнуляем EDX для корректной работы div
mov ebx,5 ; EBX=5
div ebx ; EAX=EAX/3, EDX=остаток от деления
mov edi,eax ; запись результата вычисления в 'edi'
; ---- Вывод результата на экран
```

рис.14



```
vvkopihlova@dk3n55 ~/work/arch-pc/lab06 $ nasm -f elf lab6-3.asm
vvkopihlova@dk3n55 ~/work/arch-pc/lab06 $ ld -m elf_i386 -o lab6-3 lab6-3.o
vvkopihlova@dk3n55 ~/work/arch-pc/lab06 $ ./lab6-3
Результат: 5
Остаток от деления: 1
```

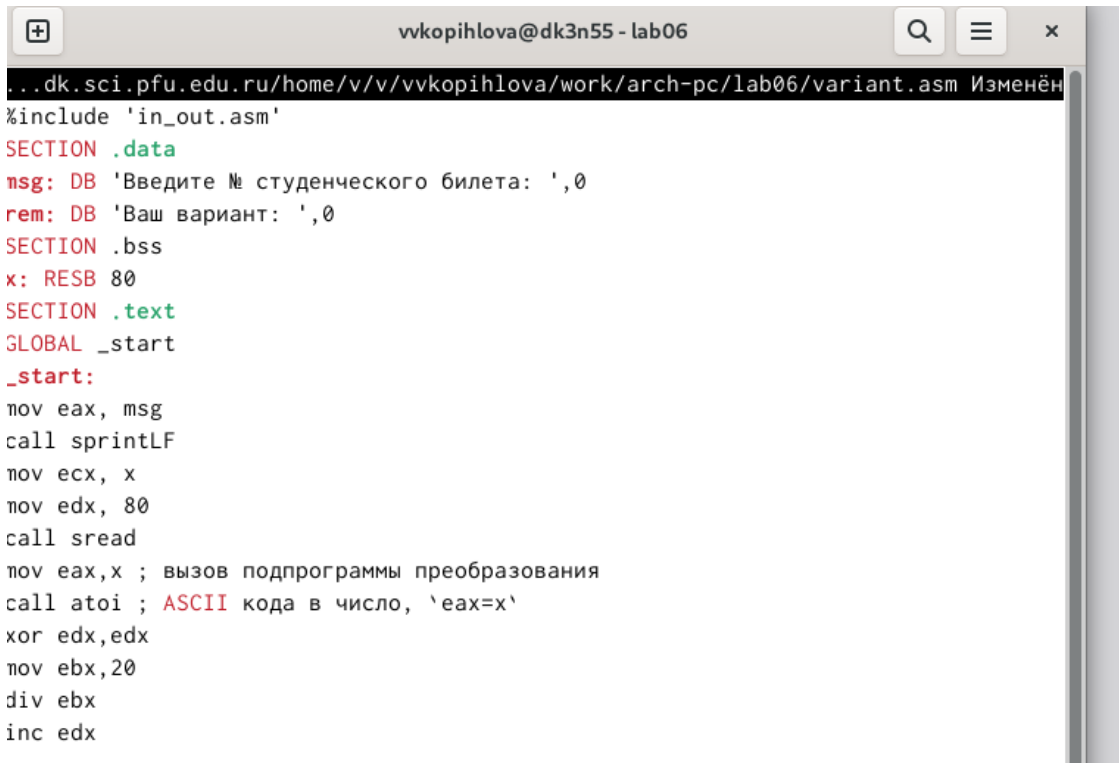
рис.15

7. В качестве другого примера рассмотрим программу вычисления варианта задания по номеру студенческого билета (рис.16 [-@fig:016]), (рис.17 [-@fig:017]), (рис.17 [-@fig:017])



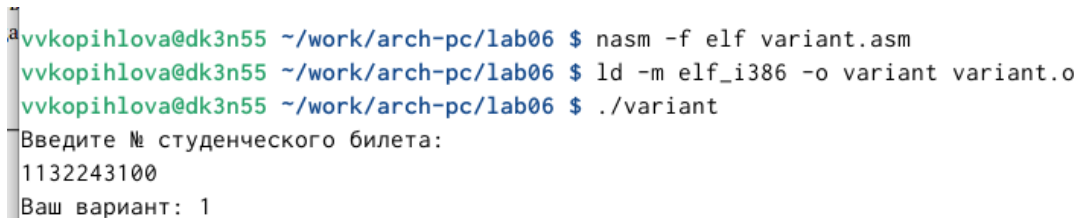
```
vvkopihlova@dk3n55 ~/work/arch-pc/lab06 $ touch variant.asm
```

рис.16



```
...dk.sci.pfu.edu.ru/home/v/v/vvkopihlova/work/arch-pc/lab06/variant.asm Изменён
#include 'in_out.asm'
SECTION .data
msg: DB 'Введите № студенческого билета: ',0
rem: DB 'Ваш вариант: ',0
SECTION .bss
x: RESB 80
SECTION .text
GLOBAL _start
_start:
mov eax, msg
call sprintf
mov ecx, x
mov edx, 80
call sread
mov eax, x ; вызов подпрограммы преобразования
call atoi ; ASCII кода в число, 'eax=x'
xor edx, edx
mov ebx, 20
div ebx
inc edx
```

рис.17



```
vkopihlova@dk3n55 ~/work/arch-pc/lab06 $ nasm -f elf variant.asm
vkopihlova@dk3n55 ~/work/arch-pc/lab06 $ ld -m elf_i386 -o variant variant.o
vkopihlova@dk3n55 ~/work/arch-pc/lab06 $ ./variant
Введите № студенческого билета:
1132243100
Ваш вариант: 1
```

рис.18

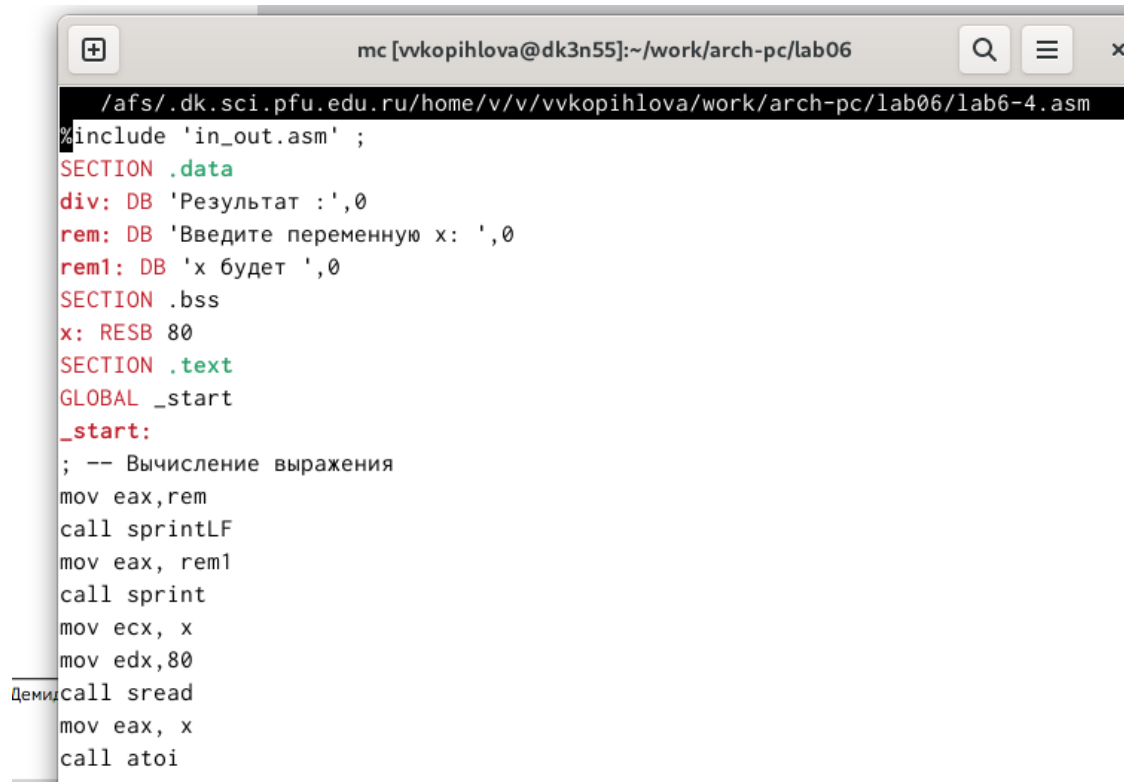
ОТВЕТЫ НА ВОПРОСЫ: 1. Какие строки листинга 6.4 отвечают за вывод на экран сообщения 'Ваш вариант: '? `mov eax, rem` `call sprint`

2. Для чего используются следующие инструкции? `mov ecx, x` - запись входной переменной в регистр `ecx` `mov edx, 80` - запись размера переменной в регистр `edx` `call sread` - вызов процедуры чтения данных
3. Для чего используется инструкция "call atoi"? функция, преобразующая ASCII код символа в целое число и записывающая результат в регистр `eax`
4. Какие строки листинга 6.4 отвечают за вычисления варианта? `xor edx, edx` `mov ebx, 20` `div ebx` `inc edx`
5. В какой регистр записывается остаток от деления при выполнении инструкции "div ebx"? `ebx`
6. Для чего используется инструкция "inc edx"? для увеличения операнда на единицу



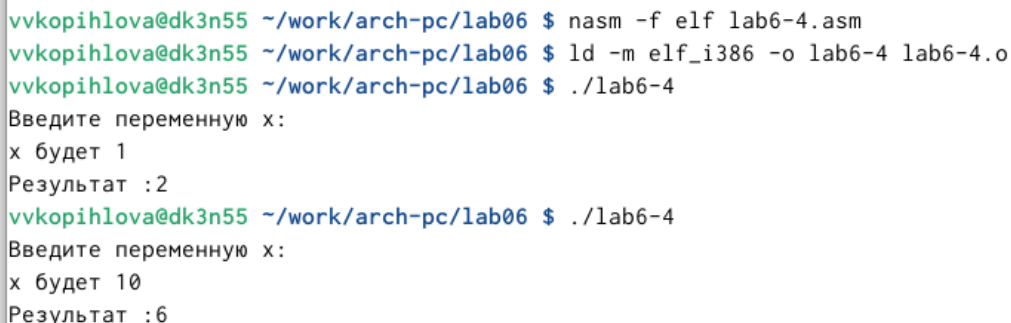
7. Какие строки листинга 6.4 отвечают за вывод на экран результата вычислений?
- ```
mov eax, rem call sprint mov eax, edx call sprintLF
```

6.4. Задание для самостоятельной работы 1. Написать программу вычисления выражения  $\square = \square(\square)$ . Программа должна выводить выражение для вычисления, выводить запрос на ввод значения  $\square$ , вычислять заданное выражение в зависимости от введенного  $\square$ , выводить результат вычислений. Вид функции  $\square(\square)$  выбрать из таблицы 6.3 вариантов заданий в соответствии с номером полученным при выполнении лабораторной работы. Создайте исполняемый файл и проверьте его работу для значений  $\square 1$  и  $\square 2$  из 6.3



```
mc [vvkopihlova@dk3n55]:~/work/arch-pc/lab06
/afs/.dk.sci.pfu.edu.ru/home/v/v/vvkopihlova/work/arch-pc/lab06/lab6-4.asm
%include 'in_out.asm' ;
SECTION .data
div: DB 'Результат : ',0
rem: DB 'Введите переменную x: ',0
rem1: DB 'x будет ',0
SECTION .bss
x: RESB 80
SECTION .text
GLOBAL _start
_start:
; -- Вычисление выражения
mov eax, rem
call sprintLF
mov eax, rem1
call sprint
mov ecx, x
mov edx, 80
call sread
mov eax, x
call atoi
```

рис.20



```
vvkopihlova@dk3n55 ~/work/arch-pc/lab06 $ nasm -f elf lab6-4.asm
vvkopihlova@dk3n55 ~/work/arch-pc/lab06 $ ld -m elf_i386 -o lab6-4 lab6-4.o
vvkopihlova@dk3n55 ~/work/arch-pc/lab06 $ ./lab6-4
Введите переменную x:
x будет 1
Результат :2
vvkopihlova@dk3n55 ~/work/arch-pc/lab06 $ ./lab6-4
Введите переменную x:
x будет 10
Результат :6
```

рис.21

## Выводы

Освоила арифметические инструкции языка ассемблера NASM.

## Список литературы