

# **Отчет по лабораторной работе № 6**

**Дисциплина: архитектура компьютеров**

Казазаев Даниил Михайлович

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>4</b>
<b>2</b>	<b>Задания Лабораторной работы</b>	<b>5</b>
<b>3</b>	<b>Выполнение лабораторной работы</b>	<b>7</b>
3.1	Ответы на вопросы . . . . .	14
<b>4</b>	<b>Выполнение самостоятельной работы</b>	<b>17</b>
<b>5</b>	<b>Вывод</b>	<b>21</b>

## Список иллюстраций

3.1	Создание файла lab6-1.asm . . . . .	7
3.2	Редактирование файла lab6-1.asm . . . . .	7
3.3	Трансляция и запуск файла lab6-1.asm . . . . .	8
3.4	Редактирование файла lab6-1.asm . . . . .	8
3.5	Трансляция и запуск файла lab6-1.asm . . . . .	8
3.6	Создание файла lab6-2.asm . . . . .	9
3.7	Редактирование файла lab6-2.asm . . . . .	9
3.8	Трансляция и запуск файла lab6-2.asm . . . . .	9
3.9	Редактирование файла . . . . .	10
3.10	Трансляция и запуск файла . . . . .	10
3.11	Редактирование файла lab6-2.asm . . . . .	11
3.12	Трансляция и запуск файла . . . . .	11
3.13	Создание файла lab6-3.asm . . . . .	11
3.14	Редактирование файла . . . . .	12
3.15	Трансляция и запуск файла . . . . .	12
3.16	Редактирую файл . . . . .	13
3.17	Трансляция и запуск файла . . . . .	13
3.18	Созадние файла variant.asm . . . . .	13
3.19	Редактирую файл . . . . .	14
3.20	Трансляция и запуск файла . . . . .	14
4.1	Созадние файла ь.asm . . . . .	17
4.2	Редактирую файл . . . . .	18
4.3	Трансляция и запуск файла . . . . .	18

# 1 Цель работы

Целью работы является освоение арифметических инструкций языка ассемблера NASM.

## 2 Задания Лабораторной работы

1. Создать файл lab6-1.asm.
2. Отредактировать файл lab6-1.asm.
3. Трансिलровать файл lab6-1.asm в объектный файл и запустить его.
4. Изменить текст программы.
5. Трансिलровать отредактированный файл lab6-1.asm в объектный файл и запустить его.
6. Создать файл lab6-2.asm.
7. Отредактировать файл lab6-2.asm.
8. Трансिलровать файл lab6-1.asm в объектный файл и запустить его.
9. Сделать аналогично файлу lab6-1.asm
10. Создать файл lab6-3.asm.
11. Отредактировать файл lab6-3.asm для вычисления функции  $(5 * 2 + 3)/3$ .
12. Трансिलровать отредактированный файл lab6-3.asm в объектный файл и запустить его.
13. Отредактировать файл lab6-3.asm для вычисления функции  $(4 * 6 + 2)/5$ .
14. Трансिलровать отредактированный файл lab6-3.asm в объектный файл и запустить его.
15. Создать файл variant.asm.
16. Отредактировать файл variant.asm, чтобы получить номер варианта для самостоятельной работы.
17. Трансिलровать файл variant.asm в объектный файл и запустить его.
18. Ответить на вопросы, связанные с файлом variant.asm. # Задания Самостоя-

тельной работы

19. Написать программу вычисления выражения  $x = x(x)$ .

### 3 Выполнение лабораторной работы

Создаю файл lab6-1.asm. (рис. [3.1])

```
dmkazaev@Ubuntu:~$ cd ~/work/study/2023-2024/Архитектура\ компьютера/arch-pc/labs/lab06
dmkazaev@Ubuntu:~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab06$ touch lab6-1.asm
dmkazaev@Ubuntu:~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab06$ ls
lab6-1.asm  presentation  report
```

Рис. 3.1: Создание файла lab6-1.asm

Редактирую файл lab6-1.asm. (рис. [3.2])



```
1 %include 'in_out.asm'
2
3 SECTION .bss
4 buf1:  RESB 80
5
6 SECTION .text
7 GLOBAL _start
8 _start:
9
10 mov eax, '6'
11 mov ebx, '4'
12 add eax, ebx
13 mov [buf1], eax
14 mov eax, buf1
15 call sprintLF
16
17 call quit
18
```

Рис. 3.2: Редактирование файла lab6-1.asm

Транслирую файл lab6-1.asm в объектный файл, после чего запускаю его. (рис. [3.3])

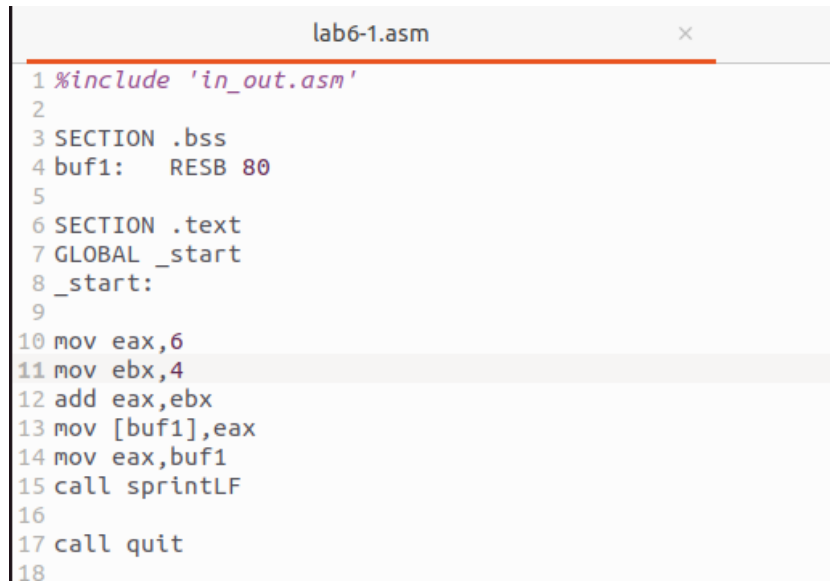
```

dnkazazaev@Ubuntu:~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab
06$ nasm -f elf lab6-1.asm
dnkazazaev@Ubuntu:~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab
06$ ld -m elf_i386 -o lab6-1 lab6-1.o
dnkazazaev@Ubuntu:~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab
06$ ./lab6-1
j

```

Рис. 3.3: Трансляция и запуск файла lab6-1.asm

Немного редактирую файл lab6-1.asm. (рис. [3.4])



```

lab6-1.asm
1 %include 'in_out.asm'
2
3 SECTION .bss
4 buf1:  RESB 80
5
6 SECTION .text
7 GLOBAL _start
8 _start:
9
10 mov eax,6
11 mov ebx,4
12 add eax,ebx
13 mov [buf1],eax
14 mov eax,buf1
15 call sprintf
16
17 call quit
18

```

Рис. 3.4: Редактирование файла lab6-1.asm

Транслирую файл lab6-1.asm в объектный файл, после чего запускаю его. (рис. [3.5])

```

06$ nasm -f elf lab6-1.asm
dnkazazaev@Ubuntu:~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab
06$ ld -m elf_i386 -o lab6-1 lab6-1.o
dnkazazaev@Ubuntu:~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab
06$ ./lab6-1

```

Рис. 3.5: Трансляция и запуск файла lab6-1.asm

Созаю файл lab6-2.asm. (рис. [3.6])



```
dmkazaev@Ubuntu: ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab
06$ touch lab6-2.asm
```

Рис. 3.6: Создание файла lab6-2.asm

Редактирую файл lab6-2.asm. (рис. [3.7])

```
lab6-1.asm
1 %include 'in_out.asm'
2
3 SECTION .text
4 GLOBAL _start
5 _start:
6
7 mov eax, '6'
8 mov ebx, '4'
9 add eax, ebx
10 call iprintLF
11
12 call quit
```

Рис. 3.7: Редактирование файла lab6-2.asm

Транслирую файл lab6-2.asm в объектный файл, после чего запускаю его. (рис. [3.8])

```
06$ touch lab6-2.asm
dmkazaev@Ubuntu: ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab
06$ nasm -f elf lab6-2.asm
dmkazaev@Ubuntu: ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab
06$ ld -m elf_i386 -o lab6-2 lab6-2.o
dmkazaev@Ubuntu: ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab
06$ ./lab6-2
106
```

Рис. 3.8: Трансляция и запуск файла lab6-2.asm

Редактирую файл. (рис. [3.9])

```

lab6-1.asm
1 %include 'in_out.asm'
2
3 SECTION .text
4 GLOBAL _start
5 _start:
6
7 mov eax,6
8 mov ebx,4
9 add eax,ebx
10 call iprintLF
11
12 call quit

```

Рис. 3.9: Редактирование файла

Транслирую файл lab6-2.asm в объектный файл, после чего запускаю его. (рис. [3.10])

```

dnkazazaev@Ubuntu:~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab
06$ nasm -f elf lab6-2.asm
dnkazazaev@Ubuntu:~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab
06$ ld -m elf_i386 -o lab6-2 lab6-2.o
dnkazazaev@Ubuntu:~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab
06$ ./lab6-2
10

```

Рис. 3.10: Трансляция и запуск файла

Меняю sprintLF на sprint в файле lab6-2.asm. (рис. [3.11])

```

1 %include 'in_out.asm'
2
3 SECTION .text
4 GLOBAL _start
5 _start:
6
7 mov eax,6
8 mov ebx,4
9 add eax,ebx
10 call iprint
11
12 call quit

```

Рис. 3.11: Редактирование файла lab6-2.asm

Транслирую файл lab6-2.asm в объектный файл, после чего запускаю его. (рис. [3.12])

```

00$ nasm -f elf lab6-2.asm
dmkazaev@Ubuntu:~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab
00$ ld -m elf_i386 -o lab6-2 lab6-2.o
dmkazaev@Ubuntu:~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab
00$ ./lab6-2
10dmkazaev@Ubuntu:~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/l
ab
00$

```

Рис. 3.12: Трансляция и запуск файла

Отличие заключается в том, что с использованием `sprintf` ввод происходит с новой строки, а при использовании `sprint` ввод продолжается на той же строке, на которой находится надпись, выведенная программой

Создаю файл lab6-3.asm. (рис. [3.13])

```

dmkazaev@Ubuntu:~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab
00$ touch lab6-3.asm

```

Рис. 3.13: Создание файла lab6-3.asm

Редактирую файл lab6-3.asm для вычисления функции  $(5 * 2 + 3)/3$ . (рис. [3.14])

```

1 ;-----
2 ; Программа вычисления выражения
3 ;-----
4
5 %include 'in_out.asm' ; подключение внешнего файла
6
7 SECTION .data
8
9 div: DB 'Результат: ',0
10 rem: DB 'Остаток от деления: ',0
11
12 SECTION .text
13 GLOBAL _start
14 _start:
15
16 ; ---- Вычисление выражения
17
18 mov eax,5 ; EAX=5
19 mov ebx,2 ; EBX=2
20 mul ebx ; EAX=EAX*EBX
21 add eax,3 ; EAX=EAX+3
22 xor edx,edx ; обнуляем EDX для корректной работы div
23 mov ebx,3 ; EBX=3
24 div ebx ; EAX=EAX/3, EDX=остаток от деления
25 mov edi,eax ; запись результата вычисления в 'edi'
26
27 ; ---- Вывод результата на экран
28
29 mov eax,div ; вызов подпрограммы печати
30 call sprint ; сообщения 'Результат: '
31 mov eax,edi ; вызов подпрограммы печати значения
32 call iprintLF ; из 'edi' в виде символов
33
34 mov eax,rem ; вызов подпрограммы печати
35 call sprint ; сообщения 'Остаток от деления: '
36 mov eax,edx ; вызов подпрограммы печати значения
37 call iprintLF ; из 'edx' (остаток) в виде символов
38
39 call quit ; вызов подпрограммы завершения
40

```

Рис. 3.14: Редактирование файла

Транслирую файл lab6-3.asm в объектный файл, после чего запускаю его. (рис. [3.15])

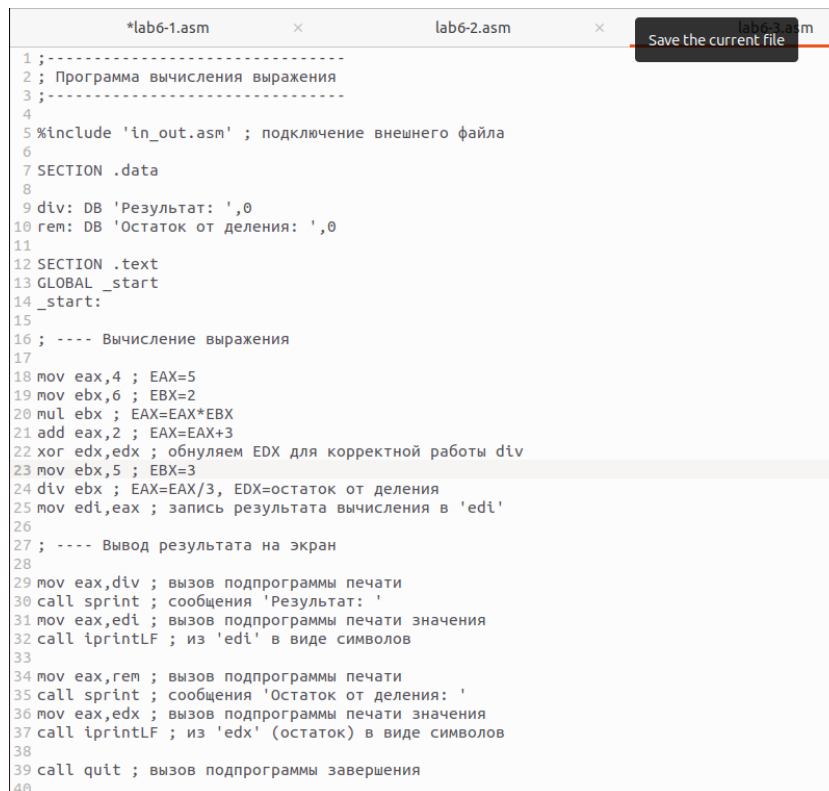
```

dmkazaev@Ubuntu: ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab
06$ nasm -f elf lab6-3.asm
dmkazaev@Ubuntu: ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab
06$ ld -m elf_i386 -o lab6-3 lab6-3.o
dmkazaev@Ubuntu: ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab
06$ ./lab6-3
Результат: 4
Остаток от деления: 1

```

Рис. 3.15: Трансляция и запуск файла

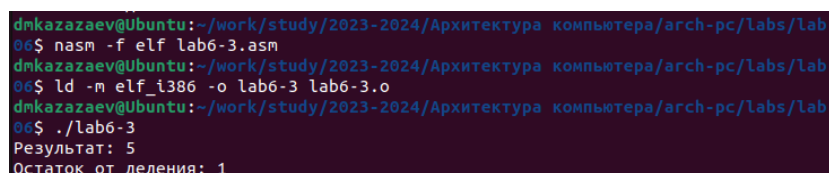
Редактирую файл lab6-3.asm для вычисления функции  $(4 * 6 + 2)/5$ . (рис. [3.16])



```
1 ;-----
2 ; Программа вычисления выражения
3 ;-----
4
5 %include 'in_out.asm' ; подключение внешнего файла
6
7 SECTION .data
8
9 div: DB 'Результат: ',0
10 rem: DB 'Остаток от деления: ',0
11
12 SECTION .text
13 GLOBAL _start
14 _start:
15
16 ; ---- Вычисление выражения
17
18 mov eax,4 ; EAX=5
19 mov ebx,6 ; EBX=2
20 mul ebx ; EAX=EAX*EBX
21 add eax,2 ; EAX=EAX+3
22 xor edx,edx ; обнуляем EDX для корректной работы div
23 mov ebx,5 ; EBX=3
24 div ebx ; EAX=EAX/3, EDX=остаток от деления
25 mov edi,eax ; запись результата вычисления в 'edi'
26
27 ; ---- Вывод результата на экран
28
29 mov eax,div ; вызов подпрограммы печати
30 call sprint ; сообщения 'Результат: '
31 mov eax,edi ; вызов подпрограммы печати значения
32 call iprintLF ; из 'edi' в виде символов
33
34 mov eax,rem ; вызов подпрограммы печати
35 call sprint ; сообщения 'Остаток от деления: '
36 mov eax,edx ; вызов подпрограммы печати значения
37 call iprintLF ; из 'edx' (остаток) в виде символов
38
39 call quit ; вызов подпрограммы завершения
40
```

Рис. 3.16: Редактирую файл

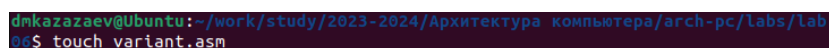
Транслирую файл lab6-3.asm в объектный файл, после чего запускаю его. (рис. [3.17])



```
dnkazazaev@Ubuntu: ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab
06$ nasm -f elf lab6-3.asm
dnkazazaev@Ubuntu: ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab
06$ ld -m elf_i386 -o lab6-3 lab6-3.o
dnkazazaev@Ubuntu: ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab
06$ ./lab6-3
Результат: 5
Остаток от деления: 1
```

Рис. 3.17: Трансляция и запуск файла

Создаю файл variant.asm. (рис. [3.18])



```
dnkazazaev@Ubuntu: ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab
06$ touch variant.asm
```

Рис. 3.18: Создание файла variant.asm

Редактирую файл variant.asm. (рис. [3.19])

```
1 ;-----
2 ; Программа вычисления варианта
3 ;-----
4
5 %include 'in_out.asm'
6
7 SECTION .data
8 msg: DB 'Введите № студенческого билета: ',0
9 rem: DB 'Ваш вариант: ',0
10
11 SECTION .bss
12 x: RESB 80
13
14 SECTION .text
15 GLOBAL _start
16 _start:
17
18 mov eax, msg
19 call sprintf
20
21 mov ecx, x
22 mov edx, 80
23 call sread
24
25 mov eax, x ; вызов подпрограммы преобразования
26 call atoi ; ASCII кода в число, 'eax=x'
27 xor edx, edx
28 mov ebx, 20
29 div ebx
30 inc edx
31
32 mov eax, rem
33 call sprintf
34 mov eax, edx
35 call sprintf
36
37 call quit
```

Рис. 3.19: Редактирую файл

Транслирую файл variant.asm в объектный файл, после чего запускаю его, чтобы получить номер варианта. (рис. [3.20])

```
dnkazazaev@Ubuntu:~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab
06$ nasm -f elf variant.asm
dnkazazaev@Ubuntu:~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab
06$ ld -m elf_i386 -o variant variant.o
dnkazazaev@Ubuntu:~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab
06$ ./variant
Введите № студенческого билета:
1132231427
Ваш вариант: 8
```

Рис. 3.20: Трансляция и запуск файла

## 3.1 Ответы на вопросы

1. Какие строки листинга 6.4 отвечают за вывод на экран сообщения 'Ваш вариант:'?

Ответ:

```
mov eax,rem  
call sprint
```

2. Для чего используются следующие инструкции?

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

Ответ: Инструкция `mov ecx, x` используется, чтобы положить адрес вводимой строки `x` в регистр `ecx`, `mov edx, 80` - запись в регистр `edx` длины вводимой строки, `call sread` - вызов подпрограммы из внешнего файла, обеспечивающей ввод сообщения с клавиатуры

3. Для чего используется инструкция “call atoi”?

Ответ: С помощью “call atoi” мы вызываем функцию `atoi`, которая преобразует `ascii`-код символа в целое число и записывает результат в регистр `eax` (перед вызовом `atoi` в регистр `eax` необходимо записать число).

4. Какие строки листинга 6.4 отвечают за вычисления варианта?

Ответ:

```
xor edx,edx  
mov ebx,20  
div ebx  
inc edx
```

5. В какой регистр записывается остаток от деления при выполнении инструкции “div ebx”?

Ответ:

Остаток записывается в регистр `edx`.

6. Для чего используется инструкция `inc edx`?

Ответ:

Для того, чтобы инкрементировать значение в регистре `edx`.

7. Какие строки листинга 6.4 отвечают за вывод на экран результата вычислений?

Ответ:

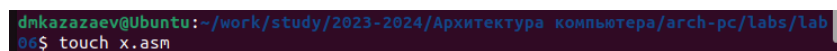
```
mov eax,edx  
call iprintLF
```



## 4 Выполнение самостоятельной работы

После выполнения прошлой программы я получил вариант 8.

Создаю файл x.asm, в котором буду выполнять задание. (рис. [4.1])



```
dmkazazev@Ubuntu:~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab  
00$ touch x.asm
```

Рис. 4.1: Создание файла x.asm

Редактирую файл x.asm. (рис. [4.2])

```
*lab6-1.asm x lab6-2.asm x *lab6-3.asm x variant.asm x
1 %include 'in_out.asm'
2
3 SECTION .data
4
5 msg: DB 'Введите значение x: ',0
6 rem: DB 'Ответ: ',0
7
8 SECTION .bss
9 x: RESB 80
10
11 SECTION .text
12 GLOBAL _start
13 _start:
14
15 mov eax,msg
16 call sprintf
17
18 mov ecx,x
19 mov edx, 80
20 call sread
21
22 mov eax, x
23 call atoi
24
25 add eax, 11
26 mov ebx, 2
27 mul ebx
28 sub eax, 6
29 mov edx,eax
30
31 mov eax,rem
32 call sprintf
33 mov eax,edx
34 call sprintf
35
36 call quit
37
```

Рис. 4.2: Редактирую файл

Транслирую файл x.asm в объектный файл, после чего запускаю его. (рис. [4.3])

```
dmkazaev@Ubuntu:~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab
06$ nasm -f elf x.asm
dmkazaev@Ubuntu:~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab
06$ ld -m elf_i386 -o x x.o
dmkazaev@Ubuntu:~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab
06$ ./x
Введите значение x:
1
Ответ: 18
dmkazaev@Ubuntu:~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab
06$ ./x
Введите значение x:
9
Ответ: 34
```

Рис. 4.3: Трансляция и запуск файла

Листинг файла x

```
%include 'in_out.asm'
```

SECTION .data

msg: DB 'Введите значение x: ',0

rem: DB 'Ответ: ',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

call atoi

add eax, 11

mov ebx, 2

mul ebx

sub eax, 6

mov edx,eax

mov eax,rem

```
call sprint
mov eax,edx
call iprintLF

call quit
```

## **5 Вывод**

При выполнении данной лабораторной работы я освоил арифметические инструкции языка ассемблера NASM.