

# **ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 6**

**дисциплина: Архитектура компьютера**

Лисовская Арина Валерьевна

# Содержание

|          |  |           |
|----------|--|-----------|
| <b>1</b> | <b>Цель работы</b>                             | <b>4</b>  |
| <b>2</b> | <b>Задание</b>                                 | <b>5</b>  |
| <b>3</b> | <b>Выполнение лабораторной работы</b>          | <b>6</b>  |
| 3.1      | Символьные и численные данные в NASM . . . . . | 6         |
| 3.2      | Задание для самостоятельной работы . . . . .   | 15        |
| <b>4</b> | <b>Выводы</b>                                  | <b>17</b> |

## Список иллюстраций

|      |                                     |    |
|------|-------------------------------------|----|
| 3.1  | создание директории . . . . .       | 6  |
| 3.2  | создания файла . . . . .            | 6  |
| 3.3  | создание копии файла . . . . .      | 6  |
| 3.4  | редактирование файла . . . . .      | 7  |
| 3.5  | запуск исполняемого файла . . . . . | 7  |
| 3.6  | редактирование файла . . . . .      | 8  |
| 3.7  | запуск исполняемого файла . . . . . | 8  |
| 3.8  | редактирование файла . . . . .      | 9  |
| 3.9  | запуск исполняемого файла . . . . . | 9  |
| 3.10 | редактирование файла . . . . .      | 10 |
| 3.11 | запуск файла . . . . .              | 10 |
| 3.12 | редактирование файла . . . . .      | 11 |
| 3.13 | запуск файла . . . . .              | 11 |
| 3.14 | создание файлов . . . . .           | 12 |
| 3.15 | редактирование файла . . . . .      | 12 |
| 3.16 | запуск файла . . . . .              | 12 |
| 3.17 | изменение программы . . . . .       | 13 |
| 3.18 | запуск файла . . . . .              | 13 |
| 3.19 | создание файлов . . . . .           | 13 |
| 3.20 | редактирование файлов . . . . .     | 14 |
| 3.21 | запуск файла . . . . .              | 14 |
| 3.22 | Редактируем файл . . . . .          | 15 |
| 3.23 | Проверяем правильность . . . . .    | 16 |
| 3.24 | Создаем копию файла . . . . .       | 16 |
| 3.25 | Самостоятельная . . . . .           | 16 |

# 1 Цель работы

Цель данной лабораторной работы - освоение арифметических инструкций языка ассемблера NASM.

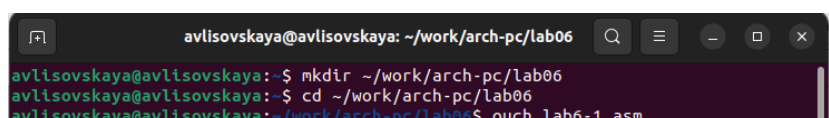
## 2 Задание

- 1.Символьные и численные данные в NASM
- 2.Выполнение арифметических операций в NASM
- 3.Выполнение заданий для самостоятельной работы

## 3 Выполнение лабораторной работы

### 3.1 Символьные и численные данные в NASM

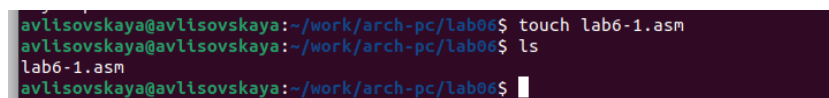
С помощью утилиты `mkdir` создаю директорию, в которой буду создавать файлы с программами для лабораторной работы №6 (рис. [3.1]). Перехожу в созданный каталог с помощью утилиты `cd`.



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

Рис. 3.1: создание директории

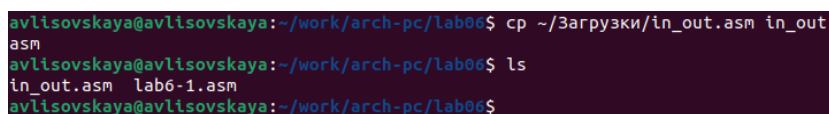
С помощью утилиты `touch` создаю файл `lab6-1.asm` (рис. [3.2]).



```
avlisovskaya@avlisovskaya:~/work/arch-pc/lab06$ touch lab6-1.asm
avlisovskaya@avlisovskaya:~/work/arch-pc/lab06$ ls
lab6-1.asm
avlisovskaya@avlisovskaya:~/work/arch-pc/lab06$
```

Рис. 3.2: создания файла

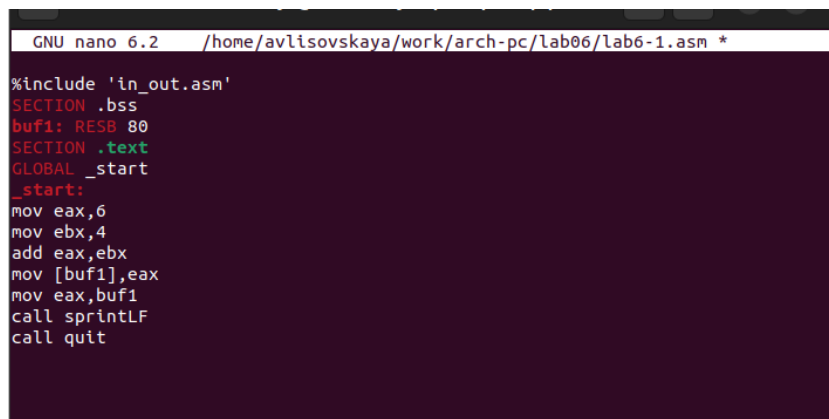
Копирую в текущий каталог файл `in_out.asm` с помощью утилиты `cp`, т.к. он будет использоваться в других программах (рис. [3.3]).



```
avlisovskaya@avlisovskaya:~/work/arch-pc/lab06$ cp ~/Загрузки/in_out.asm in_out.
asm
avlisovskaya@avlisovskaya:~/work/arch-pc/lab06$ ls
in_out.asm  lab6-1.asm
avlisovskaya@avlisovskaya:~/work/arch-pc/lab06$
```

Рис. 3.3: создание копии файла

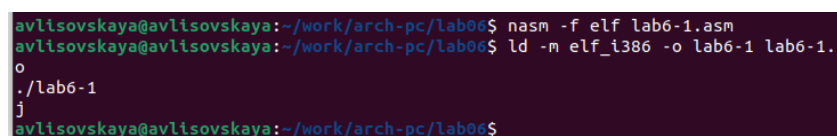
Открываю созданный файл lab6-1.asm, вставляю в него программу вывода значения регистра eax (рис. [3.4]).



```
GNU nano 6.2 /home/avlisovskaya/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 sprintf
call quit
```

Рис. 3.4: редактирование файла

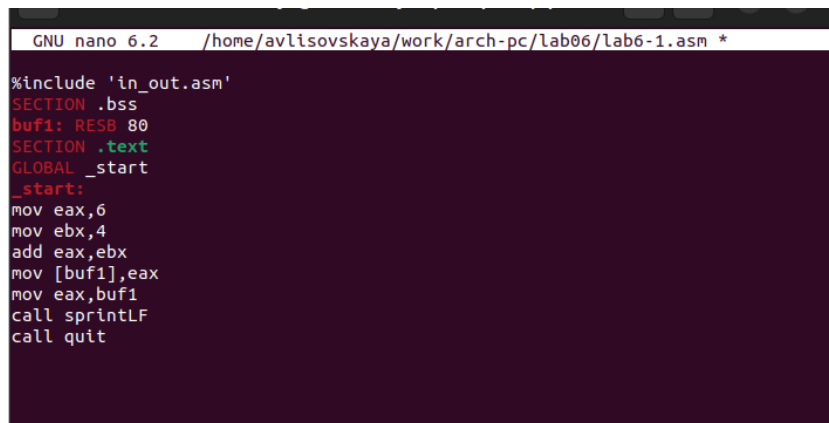
Создаю исполняемый файл программы и запускаю его (рис. [3.5]). Вывод программы: символ j, потому что программа вывела символ, соответствующий по системе ASCII сумме двоичных кодов символов 4 и 6.



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

Рис. 3.5: запуск исполняемого файла

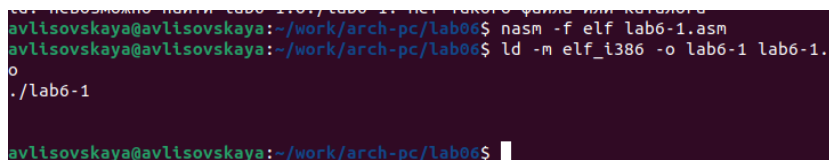
Изменяю в тексте программы символы “6” и “4” на цифры 6 и 4 (рис. [3.6]).



```
GNU nano 6.2 /home/avlisovskaya/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 sprintf
call quit
```

Рис. 3.6: редактирование файла

Создаю новый исполняемый файл программы и запускаю его (рис. [3.7]). Теперь вывелся символ с кодом 10, это символ перевода строки, этот символ не отображается при выводе на экран.



```
avlisovskaya@avlisovskaya:~/work/arch-pc/lab06$ nasm -f elf lab6-1.asm
avlisovskaya@avlisovskaya:~/work/arch-pc/lab06$ ld -m elf_i386 -o lab6-1 lab6-1.o
avlisovskaya@avlisovskaya:~/work/arch-pc/lab06$ ./lab6-1
```

Рис. 3.7: запуск исполняемого файла

Создаю новый файл lab6-2.asm с помощью утилиты touch (рис. [??]). Ввожу в файл текст другой программы для вывода значения регистра eax (рис. [3.8]).



```
GNU nano 6.2 /home/avli
%include 'in_out.asm'
SECTION .text
GLOBAL _start
_start:
mov eax,6
mov ebx,4
add eax,ebx
call iprintLF
call quit
```

Рис. 3.8: редактирование файла

Создаю и запускаю исполняемый файл lab6-2 (рис. [3.9]). Теперь вывод число 106, потому что программа позволяет вывести именно число, а не символ, хотя все еще происходит именно сложение кодов символов “6” и “4”

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

Рис. 3.9: запуск исполняемого файла

Заменяю в тексте программы в файле lab6-2.asm символы “6” и “4” на числа 6 и 4 (рис. [3.10]).

```

GLOBAL _start
_start:
    mov eax,6
    mov ebx,4
    add eax,ebx
    call iprint
    call quit

```

Рис. 3.10: редактирование файла

Создаю и запускаю новый исполняемый файл (рис. [3.11]).. Теперь программа складывает не соответствующие символам коды в системе ASCII, а сами числа, поэтому вывод 10.

```

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

```

Рис. 3.11: запуск файла

Заменяю в тексте программы функцию iprintLF на iprint (рис. [3.12]).

```

GLOBAL _start
_start:
    mov eax,6
    mov ebx,4
    add eax,ebx
    call iprint
    call quit

```

Рис. 3.12: редактирование файла

Создаю и запускаю новый исполняемый файл (рис. [3.13]). Вывод не изменился, потому что символ переноса строки не отображался, когда программа исполнялась с функцией `iprintLF`, а `iprint` не добавляет к выводу символ переноса строки, в отличие от `iprintLF`.

```

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

```

Рис. 3.13: запуск файла

### Выполнение арифметических операций в NASM

Создаю файл `lab6-3.asm` с помощью утилиты `touch` (рис. [3.14]).

```

[3.14] Установка ПК
avlisovskaya@avlisovskaya:~$ touch ~/work/arch-pc/lab06/lab6-3.asm
avlisovskaya@avlisovskaya:~$ mc

```

Рис. 3.14: создание файлов

Ввожу в созданный файл текст программы для вычисления значения выражения  $f(x) = (5 * 2 + 3)/3$  (рис. [3.15]).

```

[3.15] Редактирование файла
GNU nano 6.2 /home/avlisovskaya/work/arch-pc/lab06/lab6-3.asm *
#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/5, EDX=остаток от деления
mov edi,eax ; запись результата вычисления в 'edi'
; ---- Вывод результата на экран
mov eax,div ; вызов подпрограммы печати
call sprint ; сообщения 'Результат: '
mov eax,edi ; вызов подпрограммы печати значения
^G Справка ^O Записать ^N Поиск ^K Вырезать ^T Выполнить ^C Позиция
^X Выход ^R ЧитФайл ^\ Замена ^U Вставить ^J Выводить ^/_ К строке

```

Рис. 3.15: редактирование файла

Создаю исполняемый файл и запускаю его (рис. [3.22]).

```

[3.22] Запуск файла
avlisovskaya@avlisovskaya:~/work/arch-pc/lab06$ nasm -f elf lab6-3.asm
avlisovskaya@avlisovskaya:~/work/arch-pc/lab06$ ld -m elf_i386 -o lab6-3 lab6-3.o
avlisovskaya@avlisovskaya:~/work/arch-pc/lab06$ ./lab6-3
Результат: 5
Остаток от деления: 1
avlisovskaya@avlisovskaya:~/work/arch-pc/lab06$

```

Рис. 3.16: запуск файла

Изменяю программу так, чтобы она вычисляла значение выражения  $f(x) = (4 * 6 + 2)/5$  (рис. [3.23]).

```

GNU nano 6.2 /home/avlisovskaya/work/arch-pc/lab06/lab6-3.asm *
%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/5 EDX=остаток от деления
mov edi,eax ; запись результата вычисления в 'edi'
; ---- Вывод результата на экран
mov eax,div ; вызов подпрограммы печати
call sprint ; сообщения 'Результат: '
mov eax,edi ; вызов подпрограммы печати значения
^G Справка ^O Записать ^W Поиск ^K Вырезать ^T Выполнить ^C Позиция
^X Выход ^R ЧитФайл ^\ Замена ^U Вставить ^J Выводить ^/_ К строке

```

Рис. 3.17: изменение программы

Создаю и запускаю новый исполняемый файл (рис. [3.25]). Я посчитала для проверки правильности работы программы значение выражения самостоятельно, программа отработала верно.

```

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

```

Рис. 3.18: запуск файла

Создаю файл variant.asm с помощью утилиты touch (рис. [3.19]).

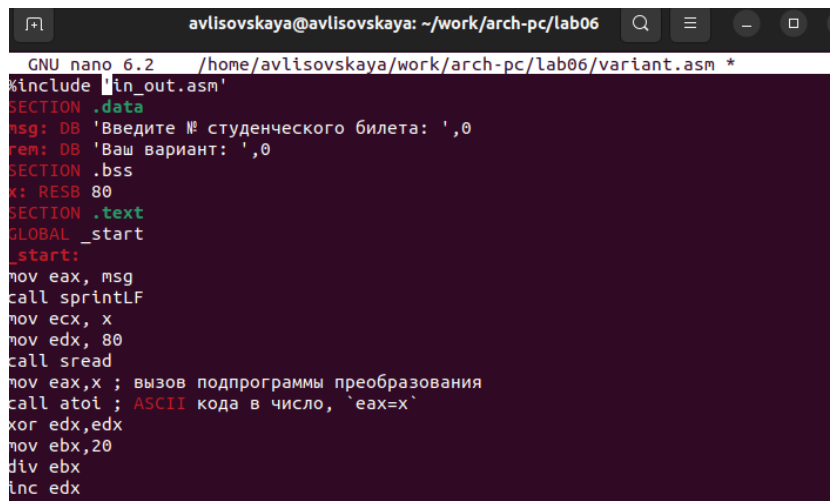
```

avlisovskaya@avlisovskaya:~/work/arch-pc/lab06$ touch ~/work/arch-pc/lab06/variant.asm
avlisovskaya@avlisovskaya:~/work/arch-pc/lab06$ mc

```

Рис. 3.19: создание файлов

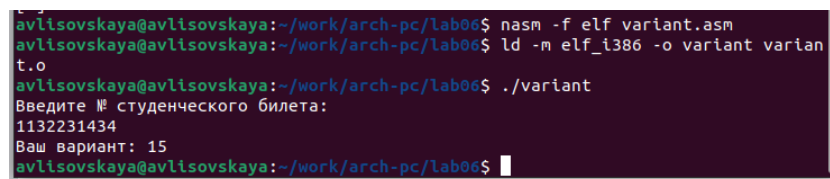
Ввожу в файл текст программы для вычисления варианта задания по номеру студенческого билета (рис. [3.20]).



```
GNU nano 6.2 /home/avlisovskaya/work/arch-pc/lab06/variant.asm *
#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. 3.20: редактирование файлов

Создаю и запускаю исполняемый файл (рис. [3.21]). Ввожу номер своего студ. билета с клавиатуры, программа вывела, что мой вариант - 8.



```
avlisovskaya@avlisovskaya:~/work/arch-pc/lab06$ nasm -f elf variant.asm
avlisovskaya@avlisovskaya:~/work/arch-pc/lab06$ ld -m elf_i386 -o variant variant.o
avlisovskaya@avlisovskaya:~/work/arch-pc/lab06$ ./variant
Введите № студенческого билета:
1132231434
Ваш вариант: 15
avlisovskaya@avlisovskaya:~/work/arch-pc/lab06$
```

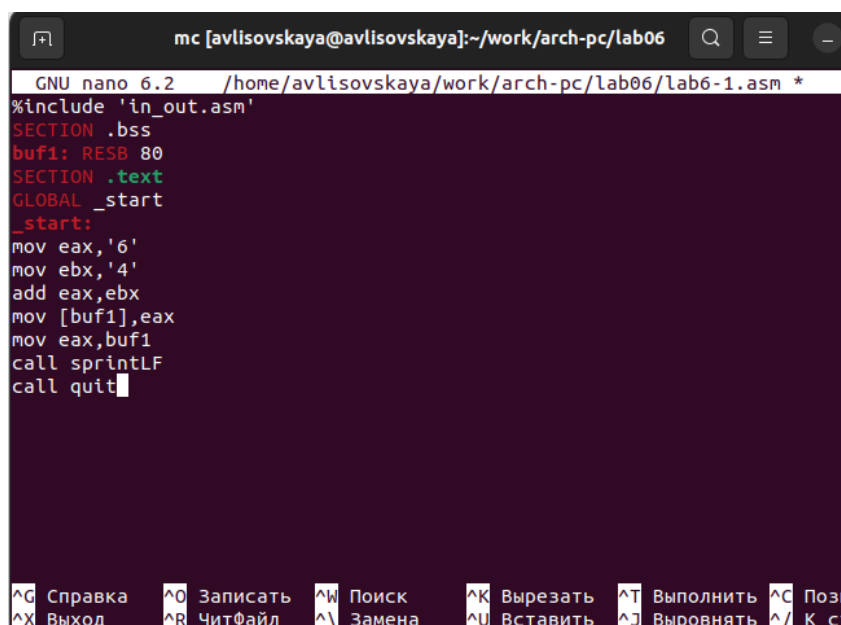
Рис. 3.21: запуск файла

Ответы на вопросы по программе 1. `oveax.rem` и строка `call sprint` отвечают за вывод на экран сообщения ‘Ваш вариант:’. 2. Эти инструкции используются для чтения строки с вводом данных от пользователя. Начальный адрес строки сохраняется в регистре `ecx`, а количество символов в строке (максимальное количество символов, которое может быть считано) сохраняется в регистре `edx`. Затем вызывается процедура `sread`, которая выполняет чтение строки. 3. Инструкция `call atoi` используется для преобразования строки в целое число. Она принимает адрес строки в регистре `eax` и возвращает полученное число в регистре `eax`. Строка `xoredx,edx` обнуляет регистр `edx` перед выполнением деления. Строка `movebx,20` загружает значение 20 в регистр `ebx`. Строка `divebx` выполняет де-

ление регистра `eax` на значение регистра `ebx` с сохранением частного в регистре `eax` и остатка в регистре `edx`, 5. Остаток от деления записывается в регистр `edx`. 6. Инструкция `inc edx` используется для увеличения значения в регистре `edx` на 1. В данном случае, она увеличивает остаток от деления на 1. 13 7. Строка `mov eax,edx` передает значение остатка от деления в регистр `eax`. 36 Строка `call iprintLF` вызывает процедуру `iprintLF` для вывода значения на экран вместе с переводом строки.

## 3.2 Задание для самостоятельной работы

Создаю файл `lab6-4.asm` с помощью утилиты `touch` (рис. [??]).



```

GNU nano 6.2 /home/avlisovskaya/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. 3.22: Редактируем файл

Открываю созданный файл для редактирования, ввожу в него текст программы для вычисления значения выражения  $(11 + x) * 2 - 6$  (рис. [??]). Это выражение было под вариантом 8.

```

Остаток от деления: 397avlisovskaya@avlisovskaya:~/work/arch-pc/lab06$ nasm -f elf lab6-4.asm
avlisovskaya@avlisovskaya:~/work/arch-pc/lab06$ ld -m elf_i386 -o lab6-4 lab6-4.o
avlisovskaya@avlisovskaya:~/work/arch-pc/lab06$ ./lab6-4
Результат: 4
Остаток от деления: 78avlisovskaya@avlisovskaya:~/work/arch-pc/lab06$

```

Рис. 3.23: Проверяем правильность

Создаю и запускаю исполняемый файл (рис. [??]).

```

avlisovskaya@avlisovskaya:~/work/arch-pc/lab06$ nasm -f elf lab6-4.asm
avlisovskaya@avlisovskaya:~/work/arch-pc/lab06$ ld -m elf_i386 -o lab6-4 lab6-4.o
avlisovskaya@avlisovskaya:~/work/arch-pc/lab06$ ./lab6-4
Результат: 15
Остаток от деления: 397avlisovskaya@avlisovskaya:~/work/arch-pc/lab06$

```

Рис. 3.24: Создаем копию файла

Провожу еще один запуск исполняемого файла для проверки работы программы с другим значением на входе (рис. [??]). Программа отработала верно.

Программа для вычисления значения выражения  $(11 + x) * 2 - 6$ .

```

GNU nano 6.2 /home/avlisovskaya/work/arch-pc/lab06/lab6-4.asm
SECTION .bss ;
x: RESB 80 ;
SECTION .text ;
GLOBAL _start
_start:
; ---- Вычисление выражения
mov eax, msg ;
call sprint ;
mov ecx, x ;
mov edx, 80 ;
call sread ;
mov eax, x ;
call atoi ; ASCII кода в число 'eax=x'
add eax, 5 ; eax=eax+5=x+5
mov ebx, eax ;
mul ebx, eax ; EAX=EAX*EBX = (x+5)*(x+5)
;call atoi ; ASCII кода в число 'ebx=x'
;add ebx, 5 ; ebx = ebx+5= x+5
;mul ebx, EAX ; EAX = EAX*EBX=(x+5)*(x+5)
sub eax, 3 ; eax = eax-3 = (x+5)*(x+5)-3
mov edi, eax ; запись результата вычисления в 'edi'
; ---- Вывод результата на экран
mov eax, rem ; вызов подпрограммы печати
call sprint ; сообщения 'Остаток от деления: '
mov eax, edi ; вызов подпрограммы печати значения
call iprint ; из 'edi' (остаток) в виде символов
call quit ; вызов подпрограммы завершения

```

Рис. 3.25: Самостоятельная



## 4 Выводы

При выполнении данной лабораторной работы я освоила арифметические инструкции языка ассемблера NASM