

# **Лабораторная работа №6**

**Арифметические операции в NASM.**

Байрамова Гюльсабах Акифовна

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>3</b>
<b>2</b>	<b>Выполнение лабораторной работы</b>	<b>4</b>
<b>3</b>	<b>Вопросы:</b>	<b>14</b>
<b>4</b>	<b>Самостоятельная работа</b>	<b>15</b>
<b>5</b>	<b>Вывод</b>	<b>18</b>

# 1 Цель работы

Освоение арифметических инструкций языка ассемблера NASM.

## 2 Выполнение лабораторной работы

1) Я создала каталог lab07 и внутри создала файл lab7-1.asm.

```
giulsabakh@dk2n21:~/work/arch-pc$ mkdir lab06
giulsabakh@dk2n21:~/work/arch-pc$ cd lab06
giulsabakh@dk2n21:~/work/arch-pc/lab06$ pwd
/home/giulsabakh/work/arch-pc/lab06
giulsabakh@dk2n21:~/work/arch-pc/lab06$
giulsabakh@dk2n21:~/work/arch-pc/lab06$ ls
giulsabakh@dk2n21:~/work/arch-pc/lab06$ touch lab6-1.asm
giulsabakh@dk2n21:~/work/arch-pc/lab06$ ls
lab6-1.asm
giulsabakh@dk2n21:~/work/arch-pc/lab06$
```

Рис. 2.1: Создание файла lab7-1.asm

2) В файл lab7-1.asm я ввела нужный текст, создала файл и проверила его.  
Программа вывела j.

```

/home/giulsabakh/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. 2.2: Текст в файле lab7-1.asm

```

nasm: fatal: unable to open input file 'lab6-1.asm': no such file or directory
giulsabakh@dk2n21:~/work/arch-pc/lab06$ nasm -f elf lab6-1.asm
giulsabakh@dk2n21:~/work/arch-pc/lab06$ ld -m elf_i386 -o lab6-1 lab6-1.o
giulsabakh@dk2n21:~/work/arch-pc/lab06$ ./lab6-1
Введите строку:
Байрамова Гюльсабах

```

Рис. 2.3: Создание файла и проверка работы

- 3) Я изменила текст программы. Когда я сделала программу и запустила ее, она вывела невидимый символ, как и должна была.

```

#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. 2.4: Изменения программы lab7-1.asm

```

nasm: fatal: unable to open input file: lab6-1.asm: no such file or directory
giulsabakh@dk2n21:~/work/arch-pc/lab06$ nasm -f elf lab6-1.asm
giulsabakh@dk2n21:~/work/arch-pc/lab06$ ld -m elf_i386 -o lab6-1 lab6-1.o
giulsabakh@dk2n21:~/work/arch-pc/lab06$ ./lab6-1
Введите строку:
Байрамова Гюльсабах

```

Рис. 2.5: Проверка программы

4) Я создала файл lab7-2.asm и ввела в него текст программы.

```
%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
```

5) Создала исполняемый файл и запустила его. Как и предполагалось, он вывел число 106.

```
giulsabakh@dk2n21:~/work/arch-pc/lab06$ nasm -f elf lab6-2.asm
lab6-2.asm:8: warning: label alone on a line without a colon might be in error [-w+label-orphan]

giulsabakh@dk2n21:~/work/arch-pc/lab06$ ld -m elf_i386 -o lab6-2 lab6-2.o

giulsabakh@dk2n21:~/work/arch-pc/lab06$ ./lab6-2
106
```

Рис. 2.7: Запуск программы lab7-2.asm

6) Я изменила текст программы lab7-2 и запустила ее. Она вывела число 10, как и должна была.

```

lab7-2.asm [-----
#include 'in_out.asm'

SECTION .text
GLOBAL _start
_start:

mov eax,6
mov ebx,4

add eax,ebx
call iprintLF
call quit

```

Рис. 2.8: Изменения в программе lab7-2.asm

```

giulsabakh@dk2n21:~/work/arch-pc/lab06$ nasm -f elf lab6-2.asm
lab6-2.asm:5: warning: label alone on a line without a colon might be in error [-w+label-orphan]

giulsabakh@dk2n21:~/work/arch-pc/lab06$ ld -m elf_i386 -o lab6-2 lab6-2.o

giulsabakh@dk2n21:~/work/arch-pc/lab06$ ./lab6-2
10

```

Рис. 2.9: Проверка программы

7) Я заменила функцию iprintLF на iprint. Проверила файл и результат отличался в выводе данных. Когда используется команда iprintLF мы начинаем вводить команду на следующей строке, а при использовании команды iprint мы вводим следующие данные на той же строке.



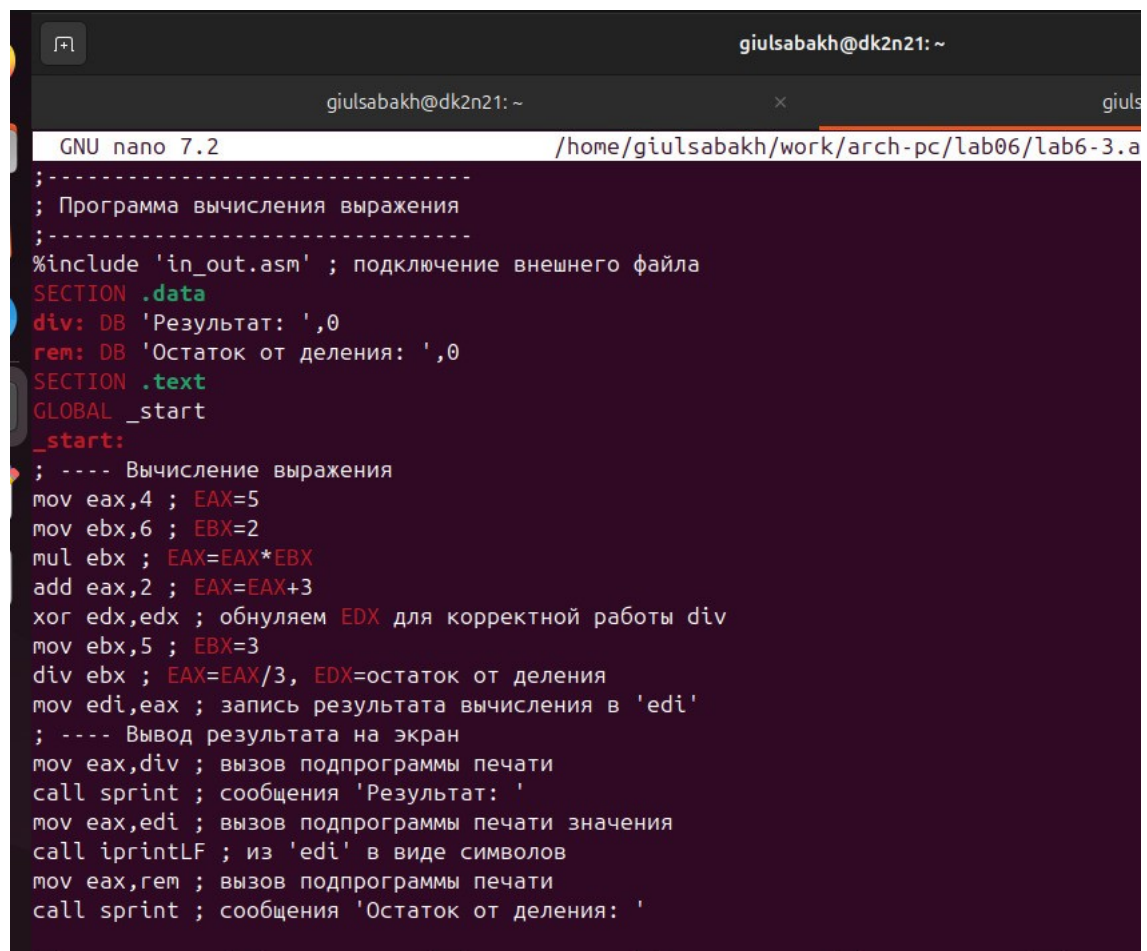
```
giulsabakh@dk2n21: ~  
GNU nano 7.2  
%include 'in_out.asm'  
  
SECTION .text  
GLOBAL _start  
_start  
  
mov eax,6  
mov ebx,4  
  
add eax,ebx  
call iprintLF  
call quit
```

Рис. 2.10: Замена iprintLF на iprint

```
giulsabakh@dk2n21:~/work/arch-pc/lab06$ nasm -f elf lab6-2.asm  
lab6-2.asm:8: warning: label alone on a line without a colon might be in error [-w+label-orphan]  
giulsabakh@dk2n21:~/work/arch-pc/lab06$ ld -m elf_i386 -o lab6-2 lab6-2.o  
giulsabakh@dk2n21:~/work/arch-pc/lab06$ ./lab6-2  
106  
i 3 1 Hello M... ^
```

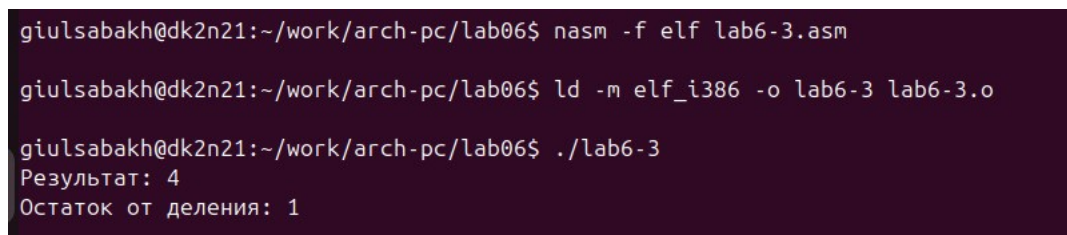
Рис. 2.11: Запуск программы с изменениями

8) Я создалф файл lab7-3.asm и ввела текст программы. Запустила и проверила работу.



```
giulsabakh@dk2n21: ~
GNU nano 7.2 /home/giulsabakh/work/arch-pc/lab06/lab6-3.a
;-----
; 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=5
mov ebx,6 ; EBX=2
mul ebx ; EAX=EAX*EBX
add eax,2 ; EAX=EAX+3
xor edx,edx ; обнуляем EDX для корректной работы div
mov ebx,5 ; EBX=3
div ebx ; EAX=EAX/3, 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
call iprintLF ; из 'edi' в виде символов
mov eax,rem ; вызов подпрограммы печати
call sprint ; сообщения 'Остаток от деления: '
```

Рис. 2.12: Текст в программе lab7-3.asm



```
giulsabakh@dk2n21:~/work/arch-pc/lab06$ nasm -f elf lab6-3.asm
giulsabakh@dk2n21:~/work/arch-pc/lab06$ ld -m elf_i386 -o lab6-3 lab6-3.o
giulsabakh@dk2n21:~/work/arch-pc/lab06$ ./lab6-3
Результат: 4
Остаток от деления: 1
```

Рис. 2.13: Запуск программы lab7-3

9) Я изменила программу для примера  $F(x) = (4*6+2)/5$ . Запустила ее и проверил

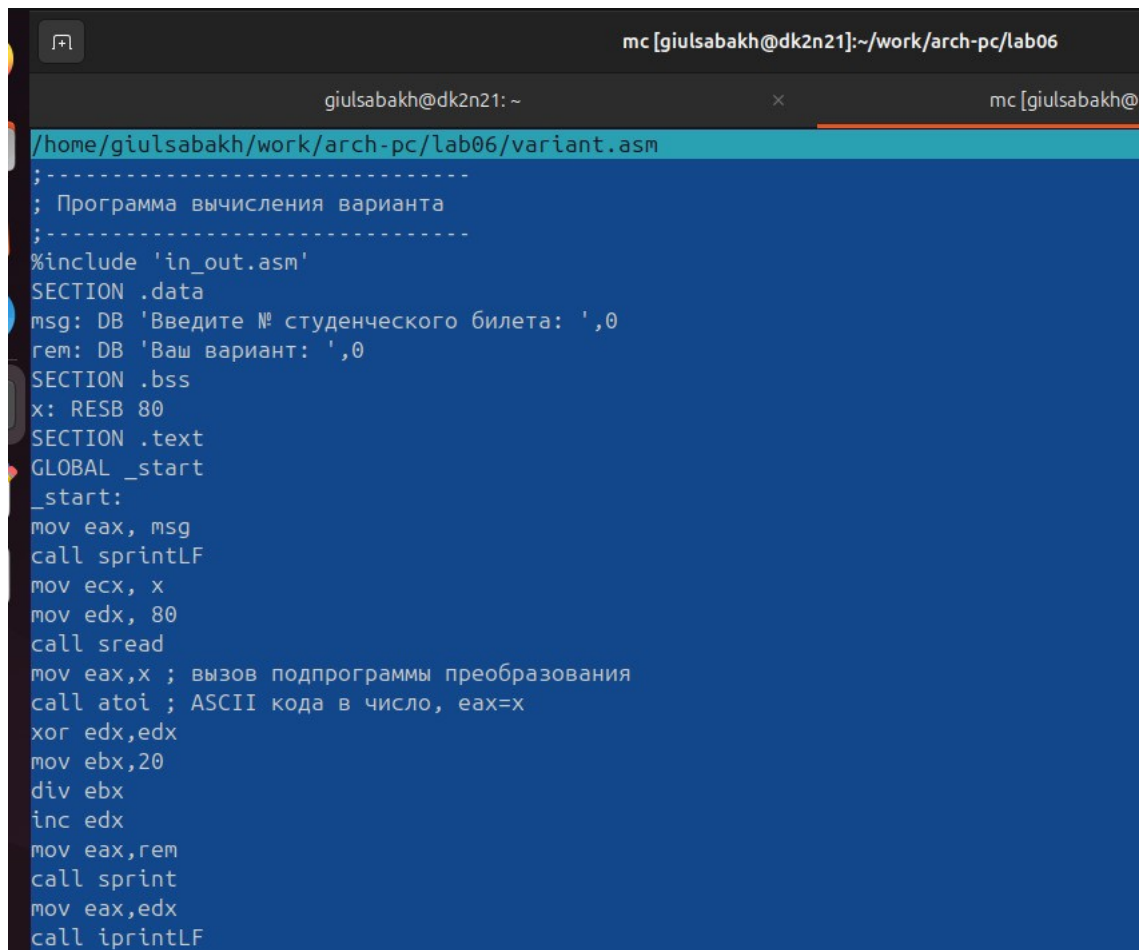
```
/home/giulsabakh/work/arch-pc/lab06/lab6-3.asm 1297/
;-----
; Программа вычисления выражения
;-----
#include 'in_out.asm' ; подключение внешнего файла
SECTION .data
div: DB 'Результат: ',0
rem: DB 'Остаток от деления: ',0
SECTION .text
GLOBAL _start
_start:
; ---- Вычисление выражения
mov eax,4 ; EAX=5
mov ebx,6 ; EBX=2
mul ebx ; EAX=EAX*EBX
add eax,2 ; EAX=EAX+3
xor edx,edx ; обнуляем EDX для корректной работы div
mov ebx,5 ; EBX=3
div ebx ; EAX=EAX/3, 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
call iprintLF ; из 'edi' в виде символов
mov eax,rem ; вызов подпрограммы печати
call sprint ; сообщения 'Остаток от деления: '
mov eax,edx ; вызов подпрограммы печати значения
call iprintLF ; из 'edx' (остаток) в виде символов
```

Рис. 2.14: Изменения в программе lab7-3

```
giulsabakh@dk2n21:~/work/arch-pc/lab06$ nasm -f elf lab6-3.asm
giulsabakh@dk2n21:~/work/arch-pc/lab06$ ld -m elf_i386 -o lab6-3 lab6-3.o
giulsabakh@dk2n21:~/work/arch-pc/lab06$ ./lab6-3
Результат: 5
Остаток от деления: 1
```

Рис. 2.15: Запуск программы

10) Я создала файл variant.asm и ввела туда нужный текст.



```
mc [giulsabakh@dk2n21]:~/work/arch-pc/lab06
giulsabakh@dk2n21: ~
/home/giulsabakh/work/arch-pc/lab06/variant.asm
;-----
; Программа вычисления варианта
;-----
#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
mov eax, rem
call sprintf
mov eax, edx
call iprintLF
```

Рис. 2.16: Текст программы в файле variant.asm

11) Я ввела в эту программу свой студенческий билет. Он вывел ответ 11, я проверила это аналитически, ответ совпал.

```
giulsabakh@dk2n21:~/work/arch-pc/lab06$ ld -m elf_i386 -o variant variant.o  
giulsabakh@dk2n21:~/work/arch-pc/lab06$ ./variant  
Введите № студенческого билета:  
1032240490  
Ваш вариант: 11
```

Рис. 2.17: Запуск программы variant

### 3 Вопросы:

1) Какие строки листинга 7.4 отвечают за вывод на экран сообщения 'Ваш вариант:'?

```
mov eax,msg call sprintLF
```

2) Для чего используются следующие инструкции? `mov ecx, x`, `mov edx, 80`, `call sread`

Эти инструкции используются для ввода переменной X с клавиатуры и сохранения введенных данных.

3) Для чего используется инструкция "call atoi"?

Эта инструкция используется для преобразования Кода переменной ASCII в число.

4) Какие строки листинга 7.4 отвечают за вычисления варианта?

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

5) В какой регистр записывается остаток от деления при выполнении инструкции "div ebx"?

В регистре ebx.

6) Для чего используется инструкция "inc edx"?

Для увеличения значения edx на 1.

7) Какие строки листинга 7.4 отвечают за вывод на экран результата вычислений?

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

## 4 Самостоятельная работа

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

```

/home/giulsabakh/work/arch-pc/lab06/lab6-4.asm 1824/
%include 'in_out.asm' ; подключение внешнего файла
SECTION .data ; секция инициализированных данных
msg: DB 'Введите значение переменной x: ',0
rem: DB 'Результат: ',0
SECTION .bss ; секция не инициализированных данных
x: RESB 80 ; Переменная, значение которой будем вводить с клавиатуры, выделенный размер - 80 байт
SECTION .text ; Код программы
GLOBAL _start ; Начало программы
_start: ; Точка входа в программу
; ---- Вычисление выражения
mov eax, msg ; запись адреса выводимого сообщения в eax
call sprint ; вызов подпрограммы печати сообщения
mov ecx, x ; запись адреса переменной в ecx
mov edx, 80 ; запись длины вводимого значения в edx
call sread ; вызов подпрограммы ввода сообщения
mov eax, x ; вызов подпрограммы преобразования
call atoi ; ASCII кода в число, eax=x
add eax, 11; eax = eax + 11 = x + 11
mov ebx, 2 ; запись значения 2 в регистр ebx
mul ebx; EAX=EAX*EBX = (x+11)*2
add eax, -6; eax = eax - 6 = (x+11)*2 - 6
mov edi, eax ; запись результата вычисления в 'edi'
; ---- Вывод результата на экран
mov eax, rem ; вызов подпрограммы печати
call sprint ; сообщения 'Результат: '
mov eax, edi ; вызов подпрограммы печати значения
call iprint ; из 'edi' в виде символов
call quit ; вызов подпрограммы завершения

```

Рис. 4.1: Программа для вычисления записанная в файле 1.asm



```
giulsabakh@dk2n21:~$ mc
giulsabakh@dk2n21:~/work/arch-pc/lab06$ ld -m elf_i386 -o lab6-4 lab6-4.o
giulsabakh@dk2n21:~/work/arch-pc/lab06$ ./lab6-4
Введите значение переменной x: 3
Результат: 22
giulsabakh@dk2n21:~$
```

Рис. 4.2: Результаты работы этой программы

## **5 Вывод**

Я освоила арифметические инструкции языка ассемблер NASM.