

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра прикладной информатики и теории вероятностей

ОТЧЕТ

ПО ЛАБОРАТОРНОЙ РАБОТЕ № 6

дисциплина: *Архитектура компьютера*

Студент: Кондрацкая Александра Евгеньевна

Группа: НММбд-02-24

МОСКВА

2024 г.

1. Цель работы

Освоить арифметические инструкции языка ассемблера NASM.

2. Задание

1. Освоить символьные и численные данные в NASM.
2. Освоить выполнение арифметических операций в NASM.

3. Выполнение лабораторной работы

Создадим каталог для программ лабораторной работы № 6, перейдём в него и создадим файл lab6-1.asm (рис.1)

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

Рис.1 Создание файла

Введём в файл lab6-1.asm текст программы из листинга (рис. 2 и 3)

```
aekondrackaya@dk3n55 ~/work/arch-pc/lab06 $ gedit lab6-1.asm
```

Рис. 2 Открываем файл



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

Рис. 3 Вводим программу

Создадим исполняемый файл и запустите его. (рис. 4)

```

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

```

Рис. 4 Создание исполняемого файла

Далее изменим текст программы и вместо символов, запишем в регистры числа (рис. 5)

```

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

aekondrackaya@dk3n55 ~/work/arch-pc/lab06 $ █

```

Рис. 5 Изменения программы

Символ не отображается.

Создадим файл lab6-2.asm в каталоге ~/work/arch-pc/lab06 (рис. 6)

```

aekondrackaya@dk3n55 ~/work/arch-pc/lab06 $ touch lab6-2.asm
aekondrackaya@dk3n55 ~/work/arch-pc/lab06 $ █

```

Рис. 6 Создание файла

Введем в него текст программы из листинга (рис. 7 и 8)

```

aekondrackaya@dk3n55 ~/work/arch-pc/lab06 $ gedit lab6-2.asm

```

Рис. 7 Открываем файл

```

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

```

Рис. 8 Вводим программу

Создадим исполняемый файл и запустим его (рис. 9)

```

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

```

Рис. 9 Создание исполняемого файла

Аналогично предыдущему примеру изменим символы на числа. (рис. 10)

```

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

```

Рис. 10 Изменение символов

Получено число 10 в результате.

Заменяем функцию `iprintLF` на `iprint`. Создадим исполняемый файл и запустим его. (рис. 11)

```

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

```

Рис. 11 Замена функции

Изменилось расположение ответа.

Создадим файл `lab6-3.asm` в каталоге `~/work/arch-pc/lab06` (рис. 12)

```

10aekondrackaya@dk3n55 ~/work/arch-pc/lab06 $ touch lab6-3.asm
aekondrackaya@dk3n55 ~/work/arch-pc/lab06 $ █

```

Внимательно изучим текст программы из листинга 6.3 и введём в `lab6-3.asm`. (рис. 13)

```

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

```

Рис. 13 Ввод программы

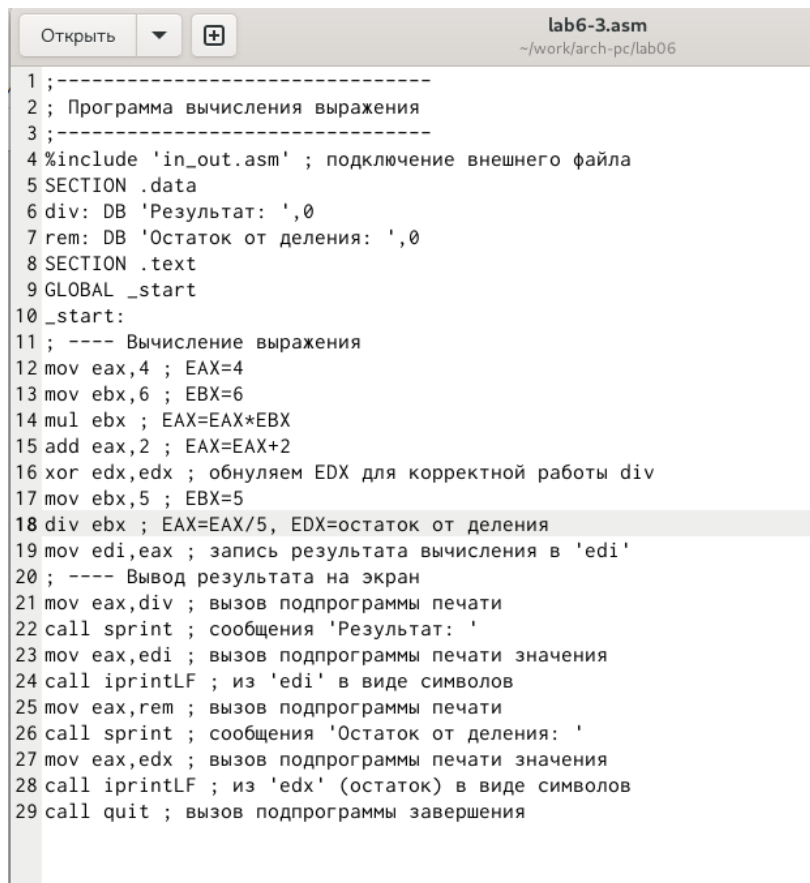
Создадим исполняемый файл и запустим его. (рис. 14)

```

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

```

Изменим текст программы для вычисления выражения $f(x) = (4 * 6 + 2)/5$. (рис. 15)

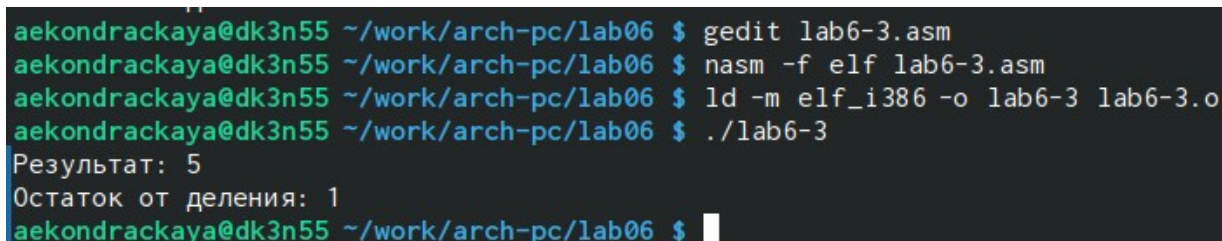


```
lab6-3.asm
~/work/arch-pc/lab06

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

Рис. 15 Изменен

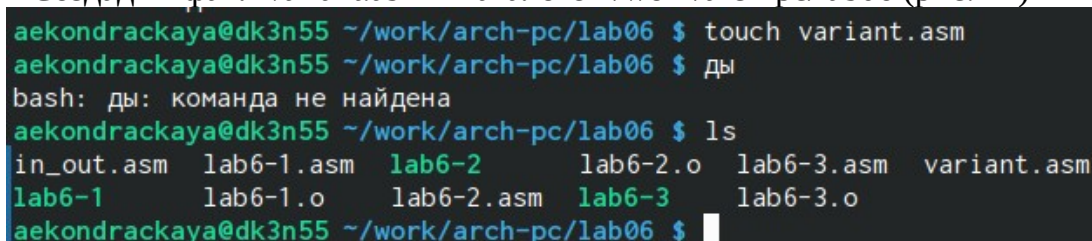
Создадим исполняемый файл и проверим его работу. (рис. 16)



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

Рис. 16 Создание исполняемого файла и проверка

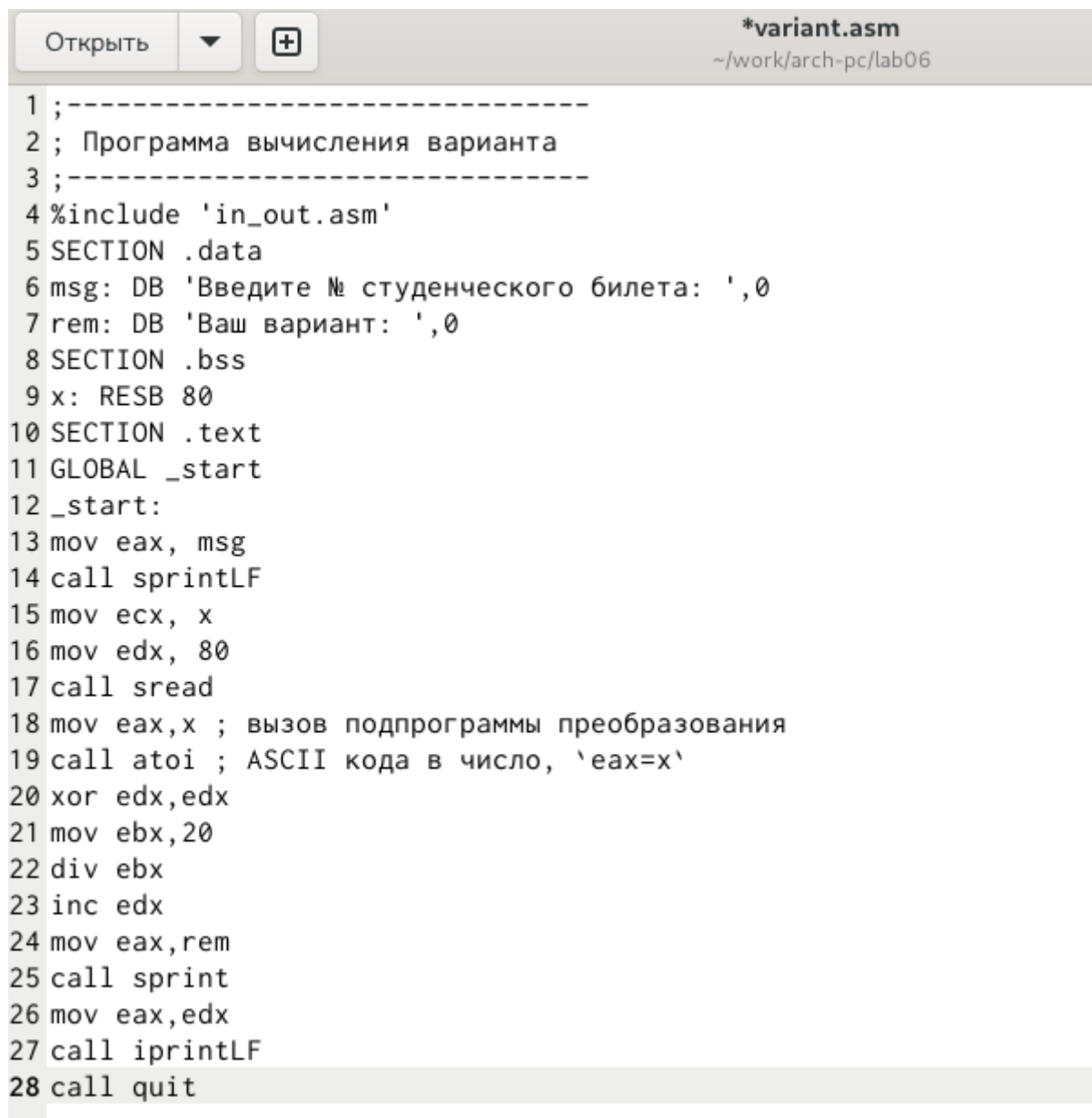
Создадим файл variant.asm в каталоге ~/work/arch-pc/lab06 (рис. 17)



```
aecondrackaya@dk3n55 ~/work/arch-pc/lab06 $ touch variant.asm
aecondrackaya@dk3n55 ~/work/arch-pc/lab06 $ ды
bash: ды: команда не найдена
aecondrackaya@dk3n55 ~/work/arch-pc/lab06 $ ls
in_out.asm  lab6-1.asm  lab6-2      lab6-2.o  lab6-3.asm  variant.asm
lab6-1      lab6-1.o    lab6-2.asm  lab6-3    lab6-3.o
aecondrackaya@dk3n55 ~/work/arch-pc/lab06 $
```

Рис. 17 Создание файла

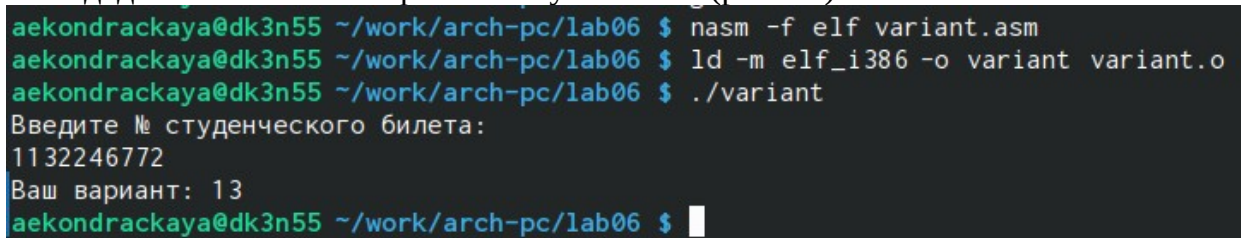
Внимательно изучим текст программы из листинга 6.4 и введём в файл variant.asm. (рис. 18)



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

Рис. 18 Ввод программы

Создадим исполняемый файл и запустим его. (рис. 19)



```
aecondrackaya@dk3n55 ~/work/arch-pc/lab06 $ nasm -f elf variant.asm
aecondrackaya@dk3n55 ~/work/arch-pc/lab06 $ ld -m elf_i386 -o variant variant.o
aecondrackaya@dk3n55 ~/work/arch-pc/lab06 $ ./variant
Введите № студенческого билета:
1132246772
Ваш вариант: 13
aecondrackaya@dk3n55 ~/work/arch-pc/lab06 $
```

Рис. 19 Создание исполняемого файла и проверка

Ответы на вопросы:

1. `mov eax,rem`
`call sprint`
2. Инструкция `mov ecx, x` используется для того, чтобы положить адрес вводимой строки `x` в регистр `ecx`. `mov edx, 80` - запись в регистр `edx` длины вводимой строки. `call sread` — это вызов подпрограммы из внешнего

- файла, обеспечивающей ввод сообщения с клавиатуры .
3. `call atoi` используется для вызова подпрограммы из внешнего файла, которая преобразует `ascii`-код символа в целое число и записывает результат в регистр `eax` .
 4. `xor edx,edx` ; обнуление `edx` для корректной работы `div`
`mov ebx,20` ; `ebx = 20`
`div ebx` ; `eax = eax/20`, `edx` - остаток от деления
`inc edx` ; `edx = edx + 1`
 5. `edx`
 6. увеличивает значение регистра `edx` на 1
 7. `mov eax,edx`
`call iprintLF`

САМОСТОЯТЕЛЬНАЯ РАБОТА

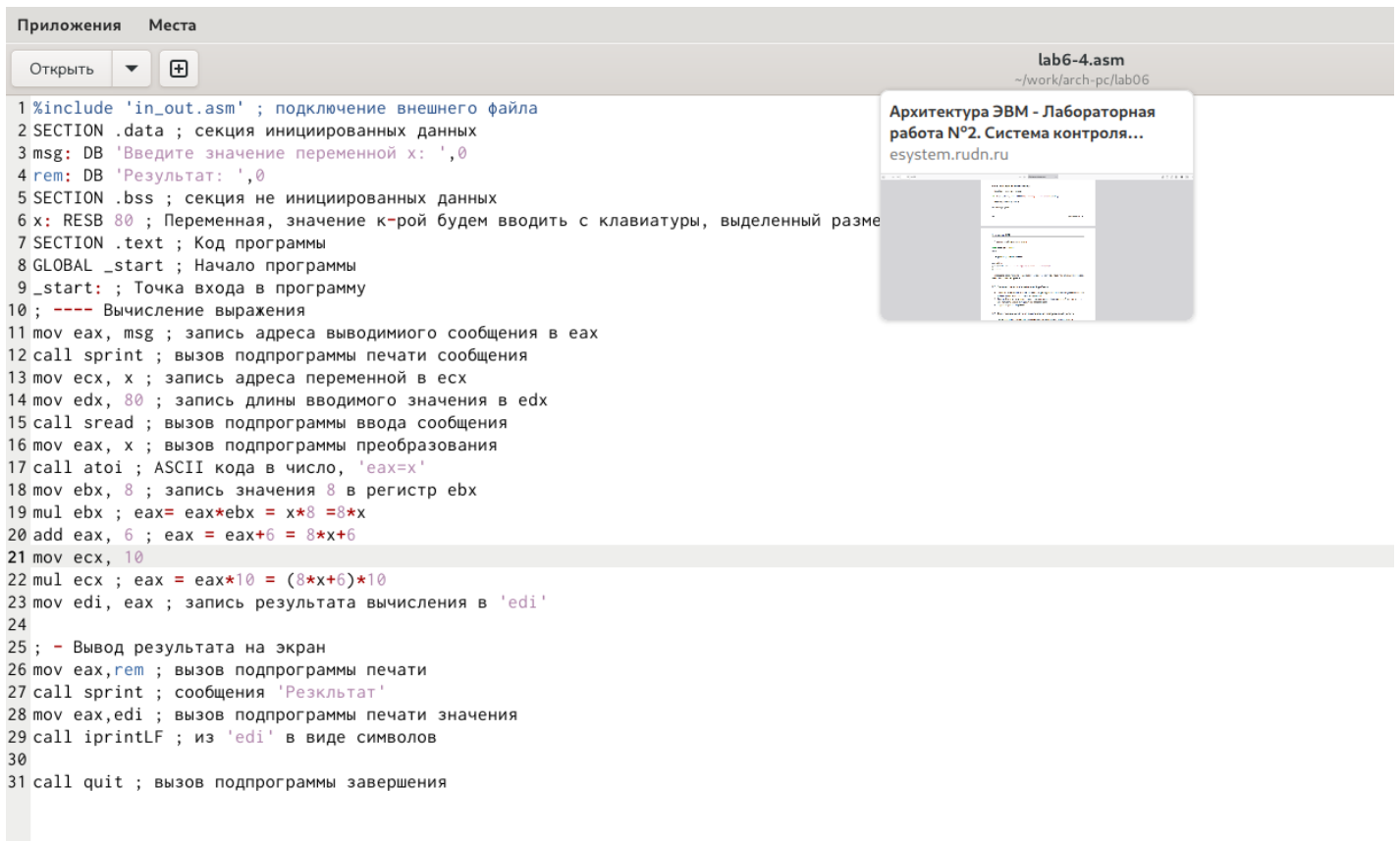


Рис. 20 Написанная программа

```
aecondrackaya@dk3n55 ~/work/arch-pc/lab06 $ gedit lab6-4.asm
aecondrackaya@dk3n55 ~/work/arch-pc/lab06 $ nasm -f elf lab6-4.asm
aecondrackaya@dk3n55 ~/work/arch-pc/lab06 $ ld -m elf_i386 -o lab6-4 lab6-4.o
aecondrackaya@dk3n55 ~/work/arch-pc/lab06 $ ./lab6-4
Введите значение переменной x: 1
Результат: 140
aecondrackaya@dk3n55 ~/work/arch-pc/lab06 $ 4
bash: 4: команда не найдена
aecondrackaya@dk3n55 ~/work/arch-pc/lab06 $ nasm -f elf lab6-4.asm
aecondrackaya@dk3n55 ~/work/arch-pc/lab06 $ ld -m elf_i386 -o lab6-4 lab6-4.o
aecondrackaya@dk3n55 ~/work/arch-pc/lab06 $ ./lab6-4
Введите значение переменной x: 4
Результат: 380
aecondrackaya@dk3n55 ~/work/arch-pc/lab06 $
```

Рис. 21 Результат выполнения

4.Выводы

Мы освоение арифметические инструкции языка ассемблера NASM.