

Отчет по лабораторной работе № 6

Арифметические операции в NASM.

Жукова София Викторовна

Содержание

Цель работы	5
Выполнение лабораторной работы	6
Выводы	17

Список иллюстраций

1	Создаем каталог и файл	6
2	Переписываем код	7
3	Запускаем файл и смотрим на его работу	7
4	Изменяем файл	8
5	Проверяем работу прораммы	8
6	Создаем файл	8
7	Заполняем файл	9
8	Смотрим на работу программы	9
9	Изменяем файл	9
10	Проверяем работу программы	10
11	Изменяем файл	10
12	Смотрим на работу программы	10
13	Создаем файл	11
14	Заполняем файл	11
15	Смотрим на результат работы программы	11
16	Редактируем код	12
17	Смотрим на результат работы программы	12
18	Создаем файл	12
19	Заполняем файл	13
20	Запускаем программу	13
21	Создаем файл	14
22	Заполняем файл	15
23	Проверяем работу программы	15
24	Проверяем работу программы	16

Список таблиц

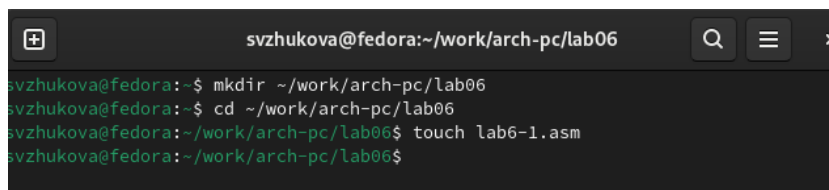
Цель работы

Освоение арифметических инструкций языка ассемблера NASM.

Выполнение лабораторной работы

Символьные и численные данные в NASM

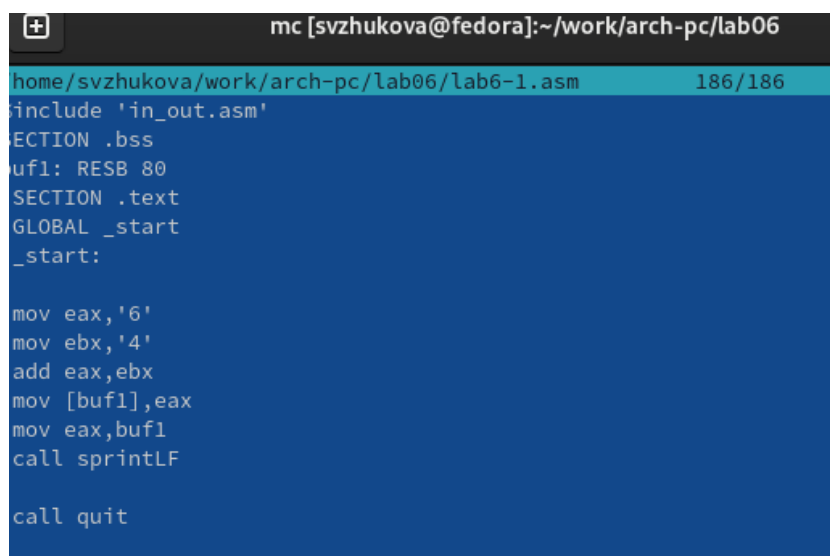
Создадим каталог для программ лабораторной работы № 6, перейдем в него и создадим файл lab6-1.asm (рис. [-@fig:001]).

A terminal window with a dark background. The title bar shows 'svzhukova@fedora:~/work/arch-pc/lab06'. The terminal contains the following commands and their outputs:

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

Рис. 1: Создаем каталог и файл

Откроем файл lab6-1.asm в Midnight Commander и заполним его в соответствии с листингом (рис. [-@fig:002]).

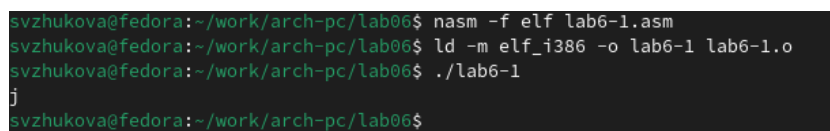


```
mc [svzhukova@fedora]:~/work/arch-pc/lab06
home/svzhukova/work/arch-pc/lab06/lab6-1.asm 186/186
include 'in_out.asm'
SECTION .bss
buf1: RESB 80
SECTION .text
GLOBAL _start
_start:

mov eax,'6'
mov ebx,'4'
add eax,ebx
mov [buf1],eax
mov eax,buf1
call sprintf
call quit
```

Рис. 2: Переписываем код

Создадим исполняемый файл и запустим его. (рис. [-@fig:003]).



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

Рис. 3: Запускаем файл и смотрим на его работу

Далее изменим текст программы и вместо символов, запишем в регистры числа.Открываем файл для редактирования и убираем кавычки с числовых значений (рис. [-@fig:004]).

```

mov eax,6
mov ebx,4
add eax,ebx
mov [buf1],eax
mov eax,buf1
call sprintf
call quit

```

Рис. 4: Изменяем файл

Создадим исполняемый файл и запустите его. (рис. [-@fig:005]).

```

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

```

Рис. 5: Проверяем работу программы

Создаем новый файл в каталоге. (рис. [-@fig:006]).

```

svzhukova@fedora:~/work/arch-pc/lab06$ touch ~/work/arch-pc/lab06/lab6-2.asm
svzhukova@fedora:~/work/arch-pc/lab06$

```

Рис. 6: Создаем файл

Заполняем файл в соответствии с листингом 6.2 (рис. [-@fig:007]).


```

lab6-2.asm [----] 12 L: [ 1+ 7 8/ 12] *(86 / 12
%include 'in_out.asm'

SECTION .text
GLOBAL _start

_start:
mov eax,'6'
mov ebx,'4'
add eax,ebx
call iprintLF

call quit

```

Рис. 7: Заполняем файл

Создаем исполняемый файл и запускаем его. (рис. [-@fig:008]).

```

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

```

Рис. 8: Смотрим на работу программы

Снова открываем файл для редактирования и убираем кавычки с числовых значений (рис. [-@fig:009]).

```

mc [svzhukova@fedora]:~/work/arch-pc/lab06
lab6-2.asm [----] 9 L: [ 1+ 6 7/ 12] *(70 / 123b) 0054 0x036
%include 'in_out.asm'

SECTION .text
GLOBAL _start

_start:
mov eax,6
mov ebx,4
add eax,ebx
call iprintLF

call quit

```

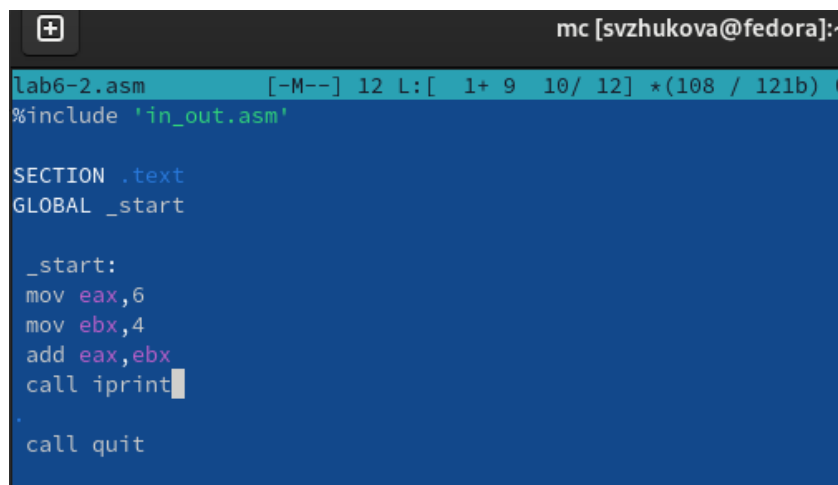
Рис. 9: Изменяем файл

Создаем исполняемый файл и запускаем его (рис. [-@fig:010]).

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

Рис. 10: Проверяем работу программы

Снова открываем файл для редактирования и меняем `iprintLF` на `iprint` (рис. [-@fig:011]).



```
lab6-2.asm [-M--] 12 L: [ 1+ 9 10/ 12] *(108 / 121b) 6
#include 'in_out.asm'

SECTION .text
GLOBAL _start

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

Рис. 11: Изменяем файл

Создаем исполняемый файл и запускаем его (рис. [-@fig:012]).

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

Рис. 12: Смотрим на работу программы

Вывод функций `iprintLF` и `iprint` отличаются только тем, что `LF` переносит на новую строку.

2. Выполнение арифметических операций в NASM

Создадим файл `lab6-3.asm` в каталоге `~/work/arch-pc/lab06` (рис. [-@fig:013]).

```
10svzhukova@fedora:~/work/arch-pc/lab06$ touch ~/work/arch-pc/lab06/lab6-3.asm
svzhukova@fedora:~/work/arch-pc/lab06$
```

Рис. 13: Создаем файл

Изучим текст программы из листинга 6.3 и введем в lab6-3.asm. (рис. [-@fig:014]).



```
mc [svzhukova@fedora]:~/work/arch-p
lab6-3.asm [----] 10 L: [ 1+ 7 8/ 26] *(144 / 318b) 0116 0x074
#include 'in_out.asm'

SECTION .data
div: DB 'Результат',0
rem: DB 'Остаток от деления',0
SECTION .text
GLOBAL _start
_start:
.
mov eax,5
mov ebx,2
mul ebx
add eax,3
xor edx,edx
mov ebx,3
div ebx
.
mov edi,eax
.
mov eax,div
call sprint
mov eax,edx
call iprintLF
.
call quit
```

Рис. 14: Заполняем файл

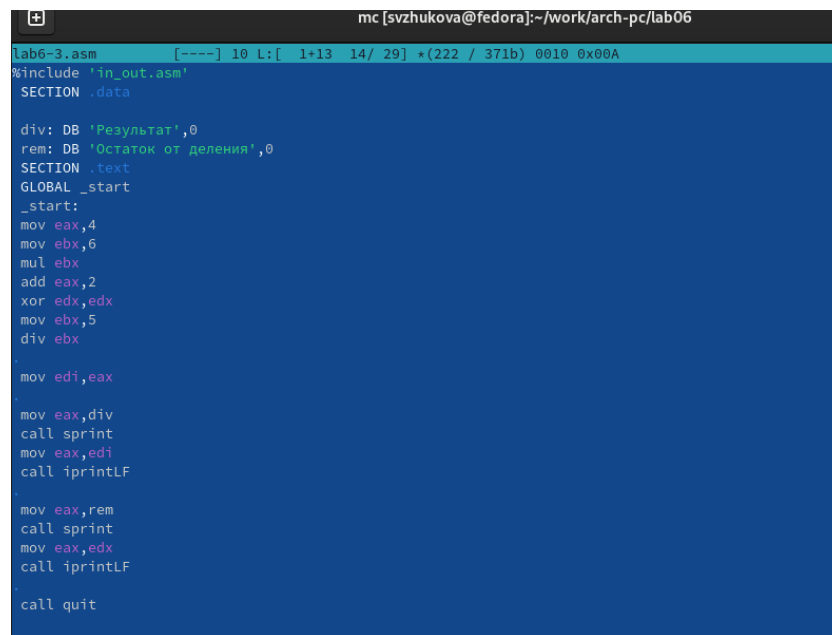
Создаем исполняемый файл и запускаем его. (рис. [-@fig:015]).

```
svzhukova@fedora:~/work/arch-pc/lab06$ nasm -f elf lab6-3.asm
svzhukova@fedora:~/work/arch-pc/lab06$ ld -m elf_i386 -o lab6-3 lab6-3.o
svzhukova@fedora:~/work/arch-pc/lab06$ ./lab6-3
Результат4
Остаток от деления1
svzhukova@fedora:~/work/arch-pc/lab06$
```

Рис. 15: Смотрим на результат работы программы

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

[-@fig:016]).



```
lab6-3.asm [----] 10 L: [ 1+13 14/ 29] *(222 / 371b) 0010 0x00A
%include 'in_out.asm'
SECTION .data

div: DB 'Результат',0
rem: DB 'Остаток от деления',0
SECTION .text
GLOBAL _start
_start:
mov eax,4
mov ebx,6
mul ebx
add eax,2
xor edx,edx
mov ebx,5
div ebx

mov edi,eax

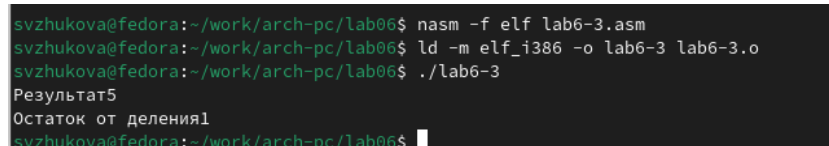
mov eax,div
call sprint
mov eax,edi
call iprintLF

mov eax,rem
call sprint
mov eax,edx
call iprintLF

call quit
```

Рис. 16: Редактируем код

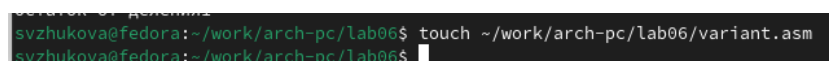
Создадим исполняемый файл и проверим его работу. (рис. [-@fig:017]).



```
svzhukova@fedora:~/work/arch-pc/lab06$ nasm -f elf lab6-3.asm
svzhukova@fedora:~/work/arch-pc/lab06$ ld -m elf_i386 -o lab6-3 lab6-3.o
svzhukova@fedora:~/work/arch-pc/lab06$ ./lab6-3
Результат5
Остаток от деления1
svzhukova@fedora:~/work/arch-pc/lab06$
```

Рис. 17: Смотрим на результат работы программы

Создадим файл variant.asm в каталоге ~/work/arch-pc/lab06 (рис. [-@fig:018]).



```
svzhukova@fedora:~/work/arch-pc/lab06$ touch ~/work/arch-pc/lab06/variant.asm
svzhukova@fedora:~/work/arch-pc/lab06$
```

Рис. 18: Создаем файл

Открываем файл и редактируем в соответствии с листингом 6.4 (рис. [-@fig:019]).

```
mc [svzhukova@fedora]:~/work/arch-pc/lab06
variant.asm [-M--] 10 L:[ 1+ 3 4/ 34] *(48 / 425b) 1042 0x412
#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 sprintLF

mov ecx, x
mov edx, 80
call sread

mov eax, x
call atoi

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

mov eax, rem
call sprint
mov eax, edx
call iprintLF

call quit
```

Рис. 19: Заполняем файл

Компилируем файл и запускаем его (рис. [-@fig:020]).

```
svzhukova@fedora:~/work/arch-pc/lab06$ nasm -f elf variant.asm
svzhukova@fedora:~/work/arch-pc/lab06$ ld -m elf_i386 -o variant.o
svzhukova@fedora:~/work/arch-pc/lab06$ ./variant
Введите № студенческого билета:
1032240966
Ваш вариант: 7
svzhukova@fedora:~/work/arch-pc/lab06$
```

Рис. 20: Запускаем программу

Ответы на вопросы к программе

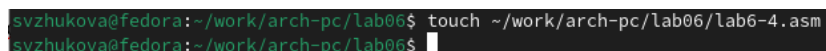
1. Какие строки листинга 6.4 отвечают за вывод на экран сообщения 'Ваш вариант: '? Строки "mov eax, rem" и "call sprint"
2. Для чего используются следующие инструкции? Эти инструкции используются для чтения строки с вводом данных от пользователя. Начальный

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

3. Для чего используется инструкция “`call atoi`”? Инструкция “`call atoi`” используется для преобразования строки в целое число. Она принимает адрес строки в регистре `eax` и возвращает полученное число в регистре `eax`.
4. Какие строки листинга 6.4 отвечают за вычисления варианта? Строка “`xor edx,edx`” обнуляет регистр `edx` перед выполнением деления. Строка “`mov ebx,20`” загружает значение 20 в регистр `ebx`. Строка “`div ebx`” выполняет деление регистра `eax` на значение регистра `ebx` с сохранением частного в регистре `eax` и остатка в регистре `edx`.
5. В какой регистр записывается остаток от деления при выполнении инструкции “`div ebx`”? Остаток от деления записывается в регистр `edx`.
6. Для чего используется инструкция “`inc edx`”? Инструкция “`inc edx`” используется для увеличения значения в регистре `edx` на 1. В данном случае, она увеличивает остаток от деления на 1.
7. Какие строки листинга 6.4 отвечают за вывод на экран результата вычислений? Строка “`mov eax,edx`” передает значение остатка от деления в регистр `eax`. Строка “`call iprintLF`” вызывает процедуру `iprintLF` для вывода значения на экран вместе с переводом строки.

Задание для самостоятельной работы

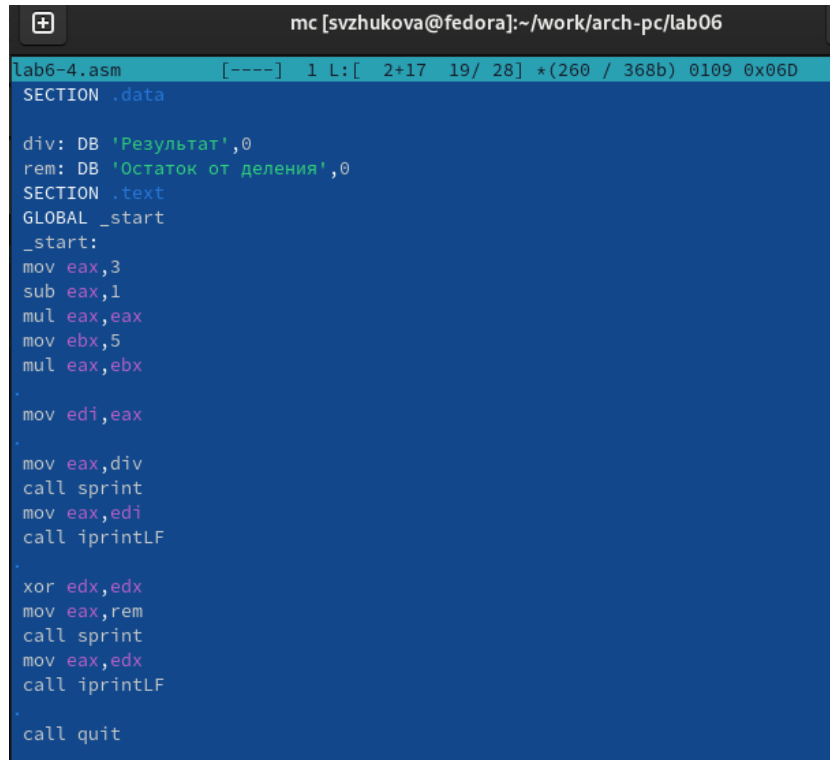
Создаем новый файл в каталоге. (рис. [-@fig:021]).



```
svzhukova@fedora: ~/work/arch-pc/lab06$ touch ~/work/arch-pc/lab06/lab6-4.asm
svzhukova@fedora: ~/work/arch-pc/lab06$
```

Рис. 21: Создаем файл

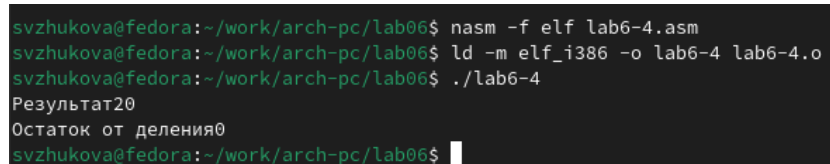
Открываем его и заполняем, чтобы решалось выражение $f(x)=5(x-1)^2$ (рис. [-@fig:022]).



```
mc [svzhukova@fedora]:~/work/arch-pc/lab06
lab6-4.asm [----] 1 L: [ 2+17 19/ 28] *(260 / 368b) 0109 0x06D
SECTION .data
div: DB 'Результат',0
rem: DB 'Остаток от деления',0
SECTION .text
GLOBAL _start
_start:
mov eax,3
sub eax,1
mul eax,eax
mov ebx,5
mul eax,ebx
mov edi,eax
mov eax,div
call sprint
mov eax,edi
call iprintLF
xor edx,edx
mov eax,rem
call sprint
mov eax,edx
call iprintLF
call quit
```

Рис. 22: Заполняем файл

Компилируем программу и проверяем для $x=3$ (рис. [-@fig:023]).



```
svzhukova@fedora:~/work/arch-pc/lab06$ nasm -f elf lab6-4.asm
svzhukova@fedora:~/work/arch-pc/lab06$ ld -m elf_i386 -o lab6-4 lab6-4.o
svzhukova@fedora:~/work/arch-pc/lab06$ ./lab6-4
Результат20
Остаток от деления0
svzhukova@fedora:~/work/arch-pc/lab06$
```

Рис. 23: Проверяем работу программы

Меняем программу для $x=1$ и проверяем как она работает (рис. [-@fig:024]).

```
svzhukova@fedora:~/work/arch-pc/lab06$ nasm -f elf lab6-4.asm
svzhukova@fedora:~/work/arch-pc/lab06$ ld -m elf_i386 -o lab6-4 lab6-4.o
svzhukova@fedora:~/work/arch-pc/lab06$ ./lab6-4
Результат0
Остаток от деления0
svzhukova@fedora:~/work/arch-pc/lab06$
```

Рис. 24: Проверяем работу программы

Выводы

Мы приобрели навыки создания исполнительных файлов для решения выражений и освоили арифметические инструкции в NASM.