

# **Отчёт по лабораторной работе**

**Лабораторная №7**

Дикач Анна Олеговна

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Выполнение лабораторной работы</b>	<b>6</b>
2.1	Символьные и численные данные NASM . . . . .	6
2.2	Выполнение арифметических операций NASM . . . . .	7
2.3	Ответы на вопросы . . . . .	8
2.4	Задание для самостоятельной работы . . . . .	9
<b>3</b>	<b>Выводы</b>	<b>11</b>

## Список иллюстраций

2.1	создание каталога и файла . . . . .	6
2.2	запуск исполняемого файла . . . . .	6
2.3	запуск исполняемого файла . . . . .	8
2.4	запуск исполняемого файла . . . . .	8
2.5	запуск исполняемого файла . . . . .	8

## **Список таблиц**

# 1 Цель работы

Освоение арифметических инструкций языка ассемблера NASM.

## 2 Выполнение лабораторной работы

### 2.1 Символьные и численные данные NASM

1. создаю каталог для программ лабораторной работы №7, перехожу в него и создаю файл lab7-1.asm (рис. 2.1)

```
[aodikach@fedora ~]$ mkdir ~/work/arch-pc/lab07
[aodikach@fedora ~]$ cd ~/work/arch-pc/lab07
[aodikach@fedora lab07]$ touch lab7-1.asm
[aodikach@fedora lab07]$ mc
```

Рис. 2.1: создание каталога и файла

2. копирую текст программы в созданный файл, также помещаю в директорию файл in\_out.asm; создаю исполняемый файл и запускаю его (рис. 2.2)

```
[aodikach@fedora lab07]$ nasm -f elf lab7-1.asm
[aodikach@fedora lab07]$ ld -m elf_i386 -o lab7-1 lab7-1.o
[aodikach@fedora lab07]$ ./lab7-1
j
[aodikach@fedora lab07]$ █
```

Рис. 2.2: запуск исполняемого файла

3. меняю текст программы, создаю исполняемый файл, запускаю его. символ на экране по таблице ASCII соответствует символу ☒ (рис. ??)(рис. ??)

```

aodikach@fedora:~/work/arch-pc/lab07
GNU nano 6.0 /home/aodikach/work/arch-pc/lab07/lab7-1.a
#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

```

```

[aodikach@fedora lab07]$ ld -m elf_i386 -o lab7-1 lab7-1.o
[aodikach@fedora lab07]$ ./lab7-1
106
[aodikach@fedora lab07]$

```

4. создаю файл lab7-2.asm и ввожу в него текст программы. создаю исполняемый файл и запускаю его (рис. ??)(рис. ??)

```

aodikach@fedora:~/work/arch-pc/lab07
GNU nano 6.0 /home/aodikach/work/arch-pc/lab07/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

```

```

[aodikach@fedora lab07]$ nasm -f elf lab7-2.asm
[aodikach@fedora lab07]$ ld -m elf_i386 -o lab7-2 lab7-2.o
[aodikach@fedora lab07]$ ./lab7-2
106

```

5. меняю текст программы, создаю исполняемый файл, запускаю его. после запуска файла получаем результат равный 10, после замены iprintLF на iprint функция не переносит ответ “10” на отдельную строку (рис. ??)(рис. ??)

```

[aodikach@fedora lab07]$ nasm -f elf lab7-2.asm
[aodikach@fedora lab07]$ ld -m elf_i386 -o lab7-2 lab7-2.o
[aodikach@fedora lab07]$ ./lab7-2
10

```

```

[aodikach@fedora lab07]$ nasm -f elf lab7-2.asm
[aodikach@fedora lab07]$ ld -m elf_i386 -o lab7-2 lab7-2.o
[aodikach@fedora lab07]$ ./lab7-2
10[aodikach@fedora lab07]$

```

## 2.2 Выполнение арифметических операций NASM

6. создаю файл lab7-3.asm, ввожу в него текст программы. создаю исполняемый файл, запускаю его (рис. 2.3)

```
[aodikach@fedora ~]$ touch ~/work/arch-pc/lab07/lab7-3.asm
[aodikach@fedora ~]$ mc

[aodikach@fedora lab07]$ nasm -f elf lab7-3.asm
[aodikach@fedora lab07]$ ld -m elf_i386 -o lab7-3 lab7-3.o
[aodikach@fedora lab07]$ ./lab7-3
Результат: 4
Остаток от деления: 1
```

Рис. 2.3: запуск исполняемого файла

7. изменяю текст программы так, чтобы вычислялась функция  $\text{f}(x) = (4 \cdot x + 2)/5$ . создаю исполняемый файл, запускаю его (рис. 2.4)

```
[aodikach@fedora lab07]$ mc

[aodikach@fedora lab07]$ nasm -f elf lab7-3.asm
[aodikach@fedora lab07]$ ld -m elf_i386 -o lab7-3 lab7-3.o
[aodikach@fedora lab07]$ ./lab7-3
Результат: 5
Остаток от деления: 1
[aodikach@fedora lab07]$ █
```

Рис. 2.4: запуск исполняемого файла

8. создаю файл variant1.asm и копирую в него текст программы. создаю исполняемый файл, запускаю его, проверяю полученный результат (рис. 2.5)

```
[aodikach@fedora lab07]$ nasm -f elf variant.asm
[aodikach@fedora lab07]$ ld -m elf_i386 -o variant variant.o
[aodikach@fedora lab07]$ ./variant
Введите № студенческого билета:
1132222009
Ваш вариант: 10
[aodikach@fedora lab07]$ █
```

Рис. 2.5: запуск исполняемого файла

## 2.3 Ответы на вопросы

1. Какие строки листинга 7.4 отвечают за вывод на экран сообщения 'Ваш вариант: '? `mov eax,rem` `call sprint`



2. Для чего используются следующие инструкции? `push mov ecx, x` `mov edx, 80`  
`call read` Благодаря этим строчкам в переменную считывается и записывается значение студенческого билета
3. Для чего используется инструкция “`call atoi`”? Эта команда преобразует `ascii`-код символа в целое число и записывает результат в регистр `eax`
4. Какие строки листинга 7.4 отвечают за вычисления варианта? Вычисление варианта происходит в строчках: `mov eax,x` `call atoi` `xor edx,edx` `mov ebx,20` `div ebx` `inc edx` (Перед этим в регистр `edx` записываем 80 (`mov edx, 80`))
5. В какой регистр записывается остаток от деления при выполнении инструкции “`div ebx`”? При выполнении инструкции `div ebx` остаток от деления запишется в регистр `edx`
6. Для чего используется инструкция “`inc edx`”? Команда `inc edx` увеличивает значение регистра `edx` на 1
7. Какие строки листинга 7.4 отвечают за вывод на экран результата вычислений? `mov eax,edx` `call iprintLF`

## 2.4 Задание для самостоятельной работы

создаю программу для вычисления функции  $5(x + 18) - 28$ . проверяю подставляя  $x_1$  и  $x_2$  (рис. ??)(рис. ??)

```

GNU nano 6.0 /home/aodikach/work/arch-p
;-----
; Программа вычисления выражения
;-----
#include 'in_out.asm'
SECTION .data
rem: DB 'f(x)=5(x + 18) - 28', 0
msg: DB 'введите переменную x: ',0

SECTION .bss
x: RESB 80

SECTION .text
GLOBAL _start
_start:
mov eax, rem
call sprintLF
mov eax, msg
call sprintLF
mov ecx, x
mov edx, 80
call sread
mov eax, x
call atoi
add eax, 18
mov ebx, 5
mul ebx
sub eax, 28
call iprintLF
call quit

```

```

[aodikach@fedora lab07]$ nasm -f elf con
[aodikach@fedora lab07]$ ld -m elf_i386
[aodikach@fedora lab07]$ ./control
f(x)=5(x + 18) - 28
введите переменную x:
2
72
[aodikach@fedora lab07]$ ./control
f(x)=5(x + 18) - 28
введите переменную x:
3
77

```

## **3 Выводы**

Освоила арифметические инструкции языка ассемблера NASM.