

# **Отчет по лабораторной работе №7**

**Дисциплина: Архитектура компьютера**

Машковцева Ксения ССergeevна, НКАбд-02-22

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Выполнение лабораторной работы</b>	<b>6</b>
<b>3</b>	<b>Выводы</b>	<b>12</b>

## Список иллюстраций

2.1	Создание папки и файла lab7.asm . . . . .	6
2.2	Запись программы в файл . . . . .	6
2.3	Запуск файла и получение неожиданного результата . . . . .	7
2.4	Запуск файла после изменения, символ не отображается . . . . .	7
2.5	Запись программы в файле lab7-2.asm . . . . .	7
2.6	Новый запуск, результат 106 . . . . .	8
2.7	Последний запуск, получен нужный результат . . . . .	8
2.8	После смены функции строка не переводится . . . . .	8
2.9	Запуск третьего файла . . . . .	8
2.10	Обновление арифметического выражения . . . . .	9
2.11	Вариант . . . . .	9
2.12	Полный текст программы . . . . .	10
2.13	Проверка программы . . . . .	11

## Список таблиц

# 1 Цель работы

Освоение арифметических инструкций языка ассемблера NASM.

## 2 Выполнение лабораторной работы

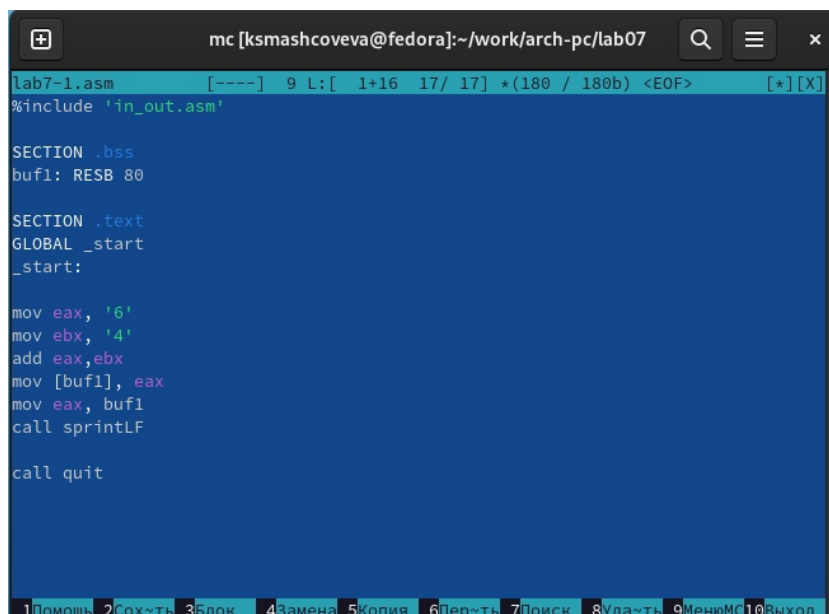
Для начала я создала каталог `~/work/arch-pc/lab07`. В нем я создаю файл `lab7.asm` (рис. 2.1).



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

Рис. 2.1: Создание папки и файла `lab7.asm`

Далее я записываю программу в файл и запускаю (рис. 2.2, 2.3).



```
lab7-1.asm  [----]  9  L: [ 1+16 17/ 17] *(180 / 180b) <EOF>  [*] [X]
#include 'in_out.asm'

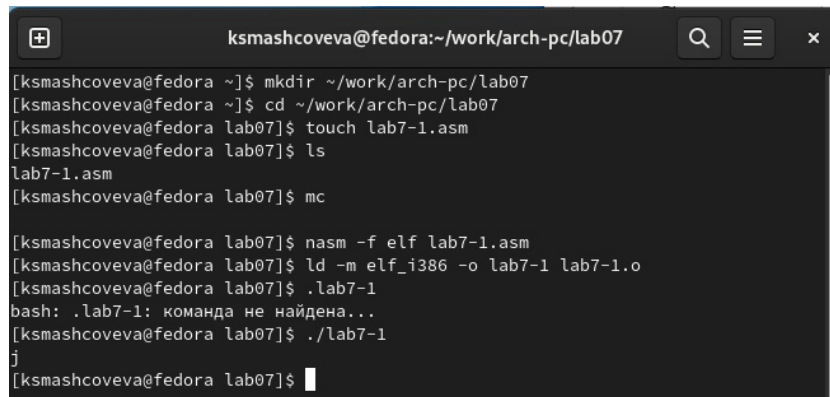
SECTION .bss
buf1: RESB 80

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: Запись программы в файл

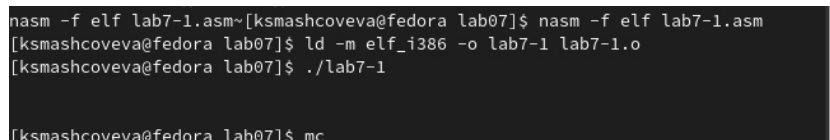


```
kasmashcoveva@fedora: ~/work/arch-pc/lab07
[kasmashcoveva@fedora ~]$ mkdir ~/work/arch-pc/lab07
[kasmashcoveva@fedora ~]$ cd ~/work/arch-pc/lab07
[kasmashcoveva@fedora lab07]$ touch lab7-1.asm
[kasmashcoveva@fedora lab07]$ ls
lab7-1.asm
[kasmashcoveva@fedora lab07]$ mc

[kasmashcoveva@fedora lab07]$ nasm -f elf lab7-1.asm
[kasmashcoveva@fedora lab07]$ ld -m elf_i386 -o lab7-1 lab7-1.o
[kasmashcoveva@fedora lab07]$ ./lab7-1
bash: ./lab7-1: команда не найдена...
[kasmashcoveva@fedora lab07]$ ./lab7-1
j
[kasmashcoveva@fedora lab07]$
```

Рис. 2.3: Запуск файла и получение неожиданного результата

Теперь вместо символов я записываю в программу числа, результат все еще не удовлетворительный. Символ не отображается (рис. 2.4).

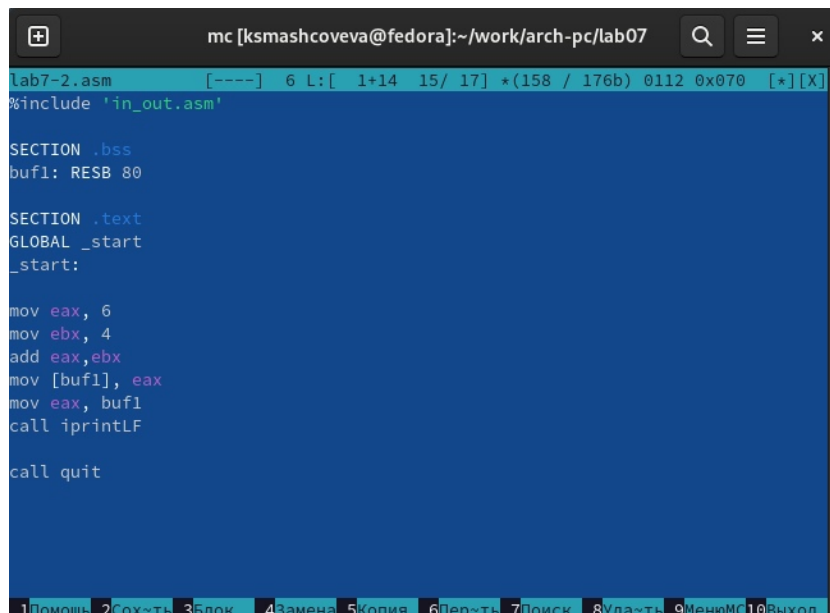


```
nasm -f elf lab7-1.asm~[kasmashcoveva@fedora lab07]$ nasm -f elf lab7-1.asm
[kasmashcoveva@fedora lab07]$ ld -m elf_i386 -o lab7-1 lab7-1.o
[kasmashcoveva@fedora lab07]$ ./lab7-1

[kasmashcoveva@fedora lab07]$ mc
```

Рис. 2.4: Запуск файла после изменения, символ не отображается

Теперь я создаю файл lab7-2.asm и записываю в него новый код (рис. 2.5).



```
lab7-2.asm  [----]  6  L: [ 1+14 15/ 17] *(158 / 176b) 0112 0x070 [*][X]
#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 iprintLF

call quit
```

Рис. 2.5: Запись программы в файле lab7-2.asm

Результат все еще не тот (рис. 2.6).

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

Рис. 2.6: Новый запуск, результат 106

Теперь в файле я меняю текстовые значения на числовые и `sprintLF` на `iprintLF`, программа наконец заработала как надо (рис. 2.7).

```
kasmashcoveva@fedora: ~/work/arch-pc/lab07
[kasmashcoveva@fedora ~]$ mc
[kasmashcoveva@fedora lab07]$ nasm -f elf lab7-2.asm
[kasmashcoveva@fedora lab07]$ ld -m elf_i386 -o lab7-2 lab7-2.o
[kasmashcoveva@fedora lab07]$ ./lab7-2
10
[kasmashcoveva@fedora lab07]$
```

Рис. 2.7: Последний запуск, получен нужный результат

После смены функции `spintLf` на `sprint` строка после выводов на экран больше не переводится (рис. 2.8).

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

Рис. 2.8: После смены функции строка не переводится

Далее я записываю код в третью программу, чтобы вычислить результат арифметического выражения и остаток от деления двух разных выражений (рис. 2.9, 2.10).

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

Рис. 2.9: Запуск третьего файла



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

Рис. 2.10: Обновление арифметического выражения

И наконец я создаю программу, которая вычисляет мой вариант для самостоятельной работы (рис. 2.11).

```
[kasmashcoveva@fedora lab07]$ nasm -f elf variant.asm
[kasmashcoveva@fedora lab07]$ ld -m elf_i386 -o variant variant.o
[kasmashcoveva@fedora lab07]$ ./variant
Введите № студенческого билета:
1132226438
Ваш вариант: 19
[kasmashcoveva@fedora lab07]$
```

Рис. 2.11: Вариант

### Ответы на вопросы

1. Строки, отвечающие за вывод сообщения:

```
mov eax,rem call sprint
```

2. nasm - используется для компиляции

mov ecx, x mov edx, 80 call sread - используются для присвоения переменной x введенного с клавиатуры значения

3. Для преобразования кода в число

4. Строки, отвечающие за вычисления варианта:

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

5. edx

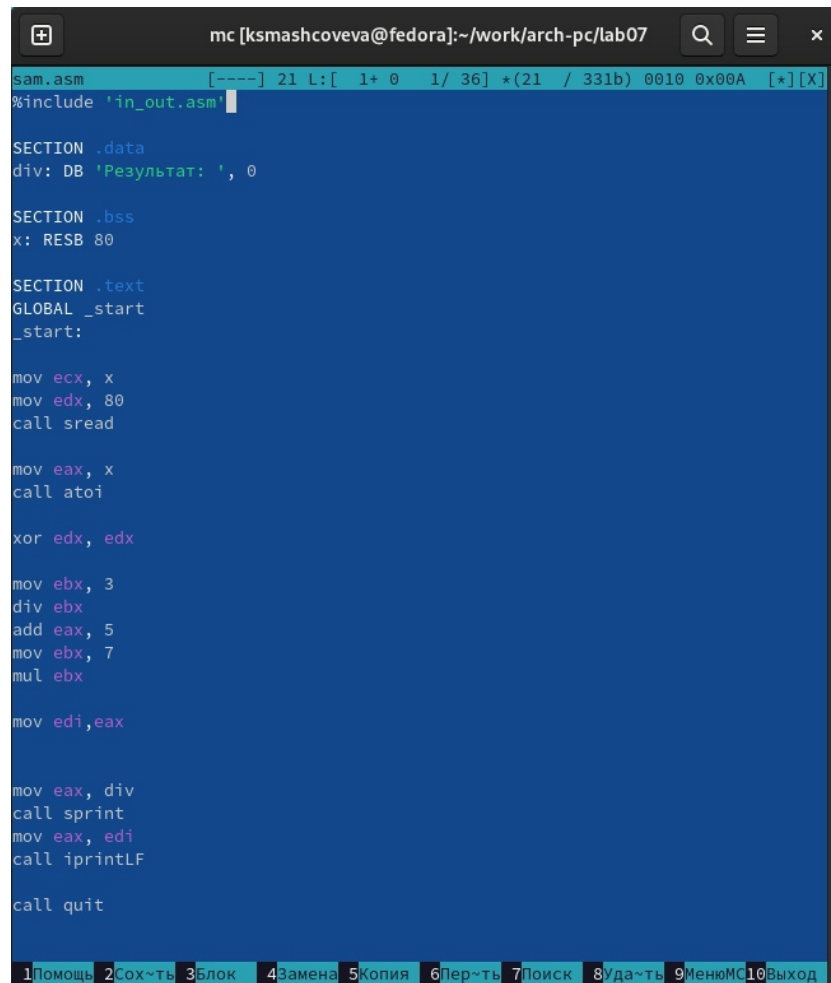
6. Строки, отвечающие за вывод варианта:

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

Задания для самостоятельной работы

Мой номер варианта - 19

Для создания программы я использовала примеры всех приведенных ранее программ и после примерно пяти попыток файл смог скомпилироваться, и на мое счастье - даже выдал верный результат! (рис. 2.12, 2.13).



```
mc [ksmashcoveva@fedora]:~/work/arch-pc/lab07
sam.asm [----] 21 L: [ 1+ 0 1/ 36] *(21 / 331b) 0010 0x00A [*][X]
#include 'in_out.asm'

SECTION .data
div: DB 'Результат: ', 0

SECTION .bss
x: RESB 80

SECTION .text
GLOBAL _start
_start:

mov ecx, x
mov edx, 80
call sread

mov eax, x
call atoi

xor edx, edx

mov ebx, 3
div ebx
add eax, 5
mov ebx, 7
mul ebx

mov edi, eax

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

call quit

1Помощь 2Сохранить 3Блок 4Замена 5Копия 6Перейти 7Поиск 8Удалить 9Меню 10Выход
```

Рис. 2.12: Полный текст программы

```
[kasmashcoveva@fedora lab07]$ nasm -f elf sam.asm
[kasmashcoveva@fedora lab07]$ ld -m elf_i386 -o sam sam.o
[kasmashcoveva@fedora lab07]$ ./sam
3
Результат: 42
[kasmashcoveva@fedora lab07]$ ./sam
9
Результат: 56
[kasmashcoveva@fedora lab07]$
```

Рис. 2.13: Проверка программы

## 3 Выводы

В этой лабораторной работе я научилась производить арифметические операции с помощью ассемблера NASM и разбираться в логических блоках других программ на данном языке.