

Отчет по лабораторной работе No7.

Арифметические операции в NASM

Коршунова Полина Юрьевна

Содержание

| | | |
|----------|--|-----------|
| 1 | Цель работы | 5 |
| 2 | Выполнение лабораторной работы | 6 |
| 3 | Ответы на вопросы | 12 |
| 4 | Выполнение задания для самостоятельной работы | 13 |
| 5 | Выводы | 15 |

Список иллюстраций

| | | |
|------|---|----|
| 2.1 | Создание каталога для программ лабораторной работы No 7 . . . | 6 |
| 2.2 | Ввожу в файл lab7-1.asm текст программы из листинга 7.1 | 6 |
| 2.3 | Проверяю результат | 7 |
| 2.4 | Изменяю текст программы | 7 |
| 2.5 | Проверяю результат | 7 |
| 2.6 | Создаю файл lab7-2.asm | 8 |
| 2.7 | Проверяю результат | 8 |
| 2.8 | Изменим символы на числа и запустим программу | 8 |
| 2.9 | Заменяю функцию iprintLF на iprint | 9 |
| 2.10 | Проверяю результат | 9 |
| 2.11 | Создаю файл lab7-3.asm | 10 |
| 2.12 | Проверяю результат | 10 |
| 2.13 | Проверяю результат | 10 |
| 2.14 | Расчитываю вариант | 11 |
| 4.1 | Пишу программу вычисления выражения по своему варианту . . | 13 |
| 4.2 | Проверяю результат | 14 |

Список таблиц

1 Цель работы

Освоение арифметических инструкций языка ассемблера NASM.

2 Выполнение лабораторной работы

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

```
pykorshunova@dk8n81 ~ $ mkdir ~/work/arch-pc/lab07
pykorshunova@dk8n81 ~ $ cd ~/work/arch-pc/lab07
pykorshunova@dk8n81 ~/work/arch-pc/lab07 $
pykorshunova@dk8n81 ~/work/arch-pc/lab07 $ touch lab7-1.asm
```

Рис. 2.1: Создание каталога для программ лабораторной работы No 7

2. Ввожу в файл lab7-1.asm текст программы из листинга 7.1. (рис. 2.2)

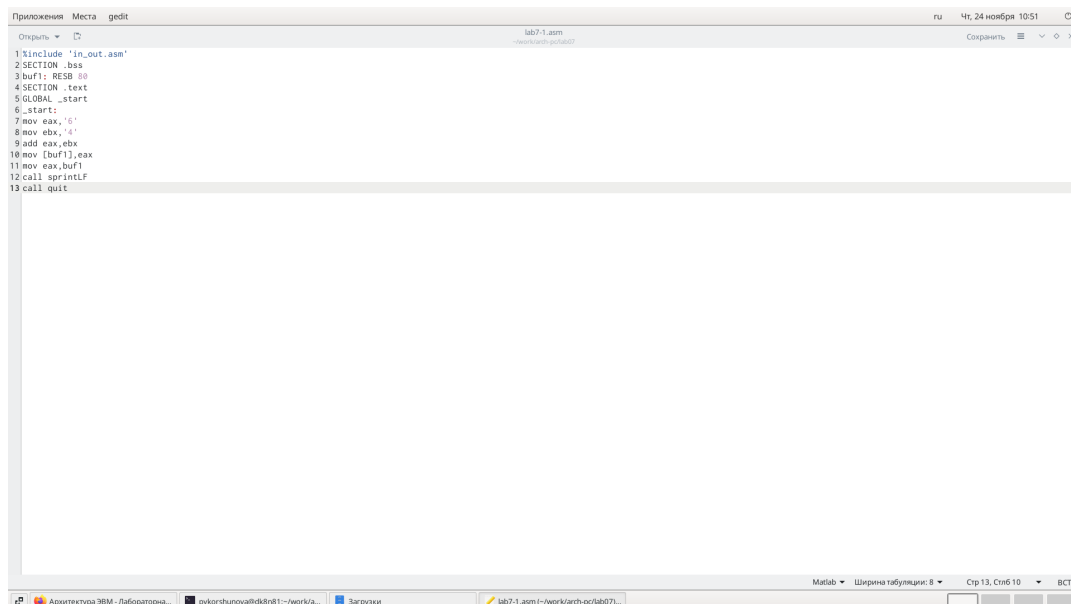


Рис. 2.2: Ввожу в файл lab7-1.asm текст программы из листинга 7.1

3. Проверяю результат (рис. 2.3)

```
pykorshunova@dk8n81 ~/work/arch-pc/lab07 $ nasm -f elf lab7-1.asm
pykorshunova@dk8n81 ~/work/arch-pc/lab07 $ ld -m elf_i386 -o lab7-1 lab7-1.o
pykorshunova@dk8n81 ~/work/arch-pc/lab07 $ ./lab7-1
j
pykorshunova@dk8n81 ~/work/arch-pc/lab07 $
```

Рис. 2.3: Проверяю результат

4. Далее изменяю текст программы и вместо символов записываю в регистры числа (рис. 2.4)

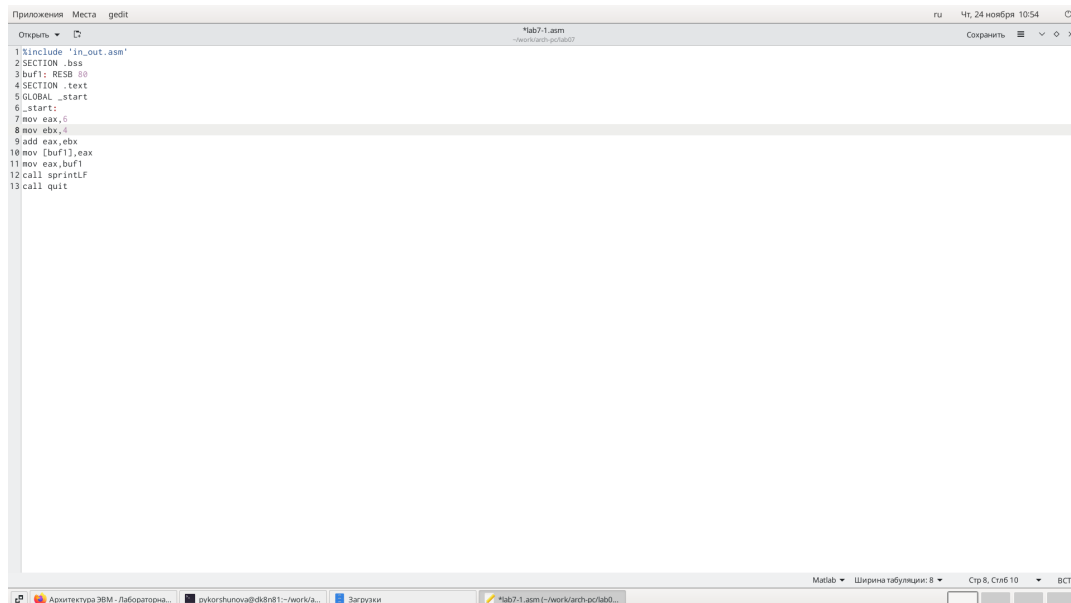


Рис. 2.4: Изменяю текст программы

5. Проверяю результат (рис. 2.5)

```
pykorshunova@dk8n81 ~/work/arch-pc/lab07 $ nasm -f elf lab7-1.asm
pykorshunova@dk8n81 ~/work/arch-pc/lab07 $ ld -m elf_i386 -o lab7-1 lab7-1.o
pykorshunova@dk8n81 ~/work/arch-pc/lab07 $ ./lab7-1

pykorshunova@dk8n81 ~/work/arch-pc/lab07 $
```

Рис. 2.5: Проверяю результат

6. Создаю файл lab7-2.asm в каталоге ~/work/arch-pc/lab07 и ввожу в него текст программы из листинга 7.2 (рис. 2.6)

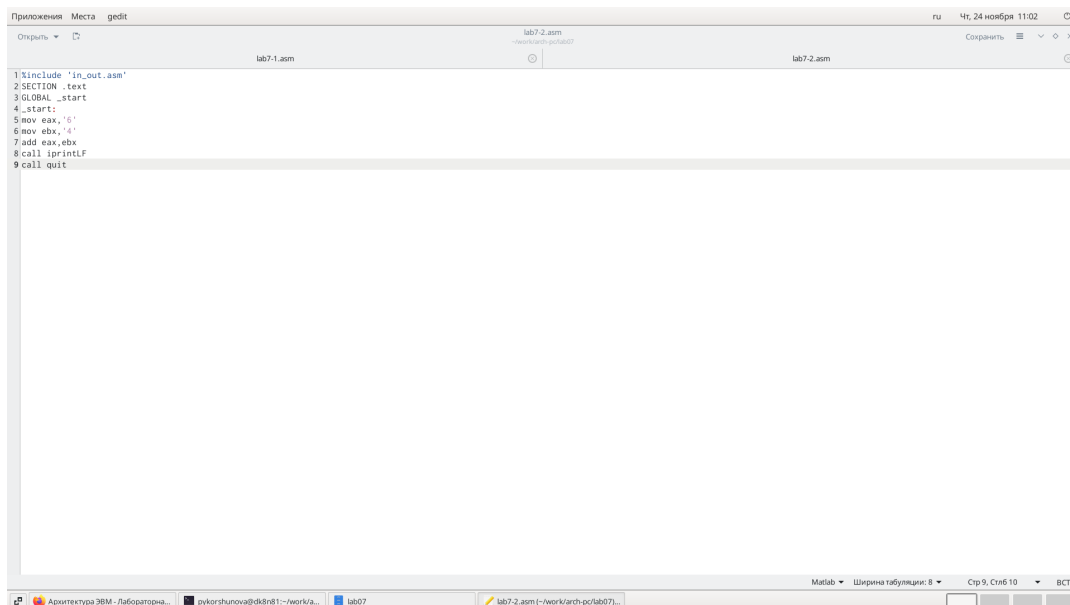


Рис. 2.6: Создаю файл lab7-2.asm

7. Проверяю результат (рис. 2.7)

```
pykorshunova@dk8n81 ~/work/arch-pc/lab07 $ touch lab7-2.asm
pykorshunova@dk8n81 ~/work/arch-pc/lab07 $ nasm -f elf lab7-2.asm
pykorshunova@dk8n81 ~/work/arch-pc/lab07 $ ld -m elf_i386 -o lab7-2 lab7-2.o
pykorshunova@dk8n81 ~/work/arch-pc/lab07 $ ./lab7-2
106
pykorshunova@dk8n81 ~/work/arch-pc/lab07 $
```

Рис. 2.7: Проверяю результат

8. Аналогично предыдущему примеру изменим символы на числа и запустим программу. Получим 10 (рис. 2.8)

```
pykorshunova@dk8n81 ~/work/arch-pc/lab07 $ nasm -f elf lab7-2.asm
pykorshunova@dk8n81 ~/work/arch-pc/lab07 $ ld -m elf_i386 -o lab7-2 lab7-2.o
pykorshunova@dk8n81 ~/work/arch-pc/lab07 $ ./lab7-2
10
pykorshunova@dk8n81 ~/work/arch-pc/lab07 $
```

Рис. 2.8: Изменим символы на числа и запустим программу

9. Заменяю функцию `iprintLF` на `iprint` (рис. 2.9)

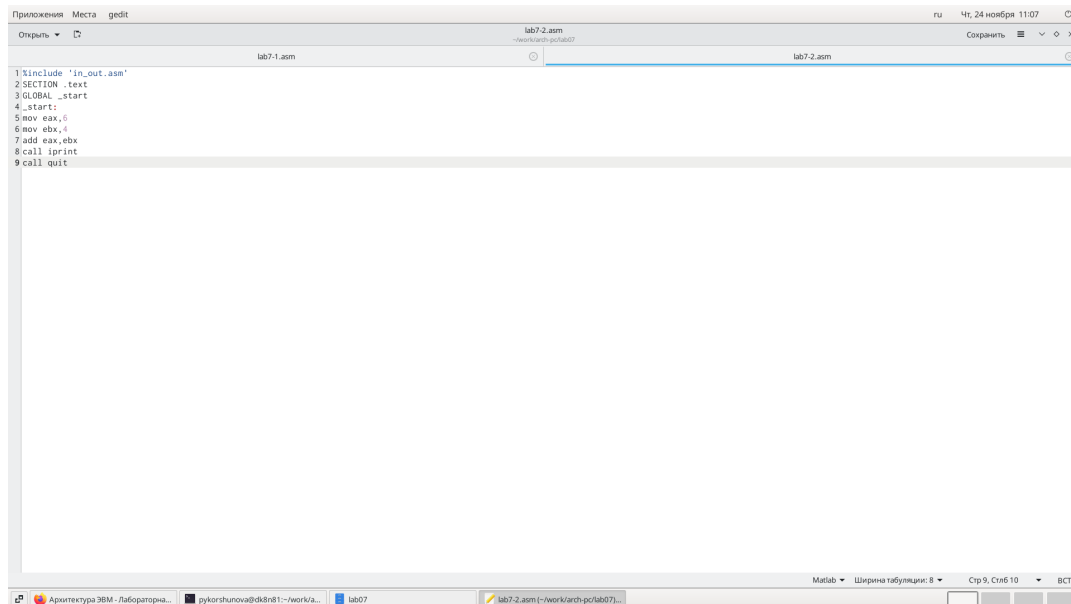


Рис. 2.9: Заменяю функцию `iprintLF` на `iprint`

10. Проверяю результат (рис. 2.10)

```
pykorshunova@dk8n81 ~/work/arch-pc/lab07 $ nasm -f elf lab7-2.asm
pykorshunova@dk8n81 ~/work/arch-pc/lab07 $ ld -m elf_i386 -o lab7-2 lab7-2.o
pykorshunova@dk8n81 ~/work/arch-pc/lab07 $ ./lab7-2
10pykorshunova@dk8n81 ~/work/arch-pc/lab07 $
```

Рис. 2.10: Проверяю результат

11. Создаю файл `lab7-3.asm` в каталоге `~/work/arch-pc/lab07` и внимательно изучаю текст программы из листинга 7.3 и ввожу в `lab7-3.asm` (рис. 2.11)

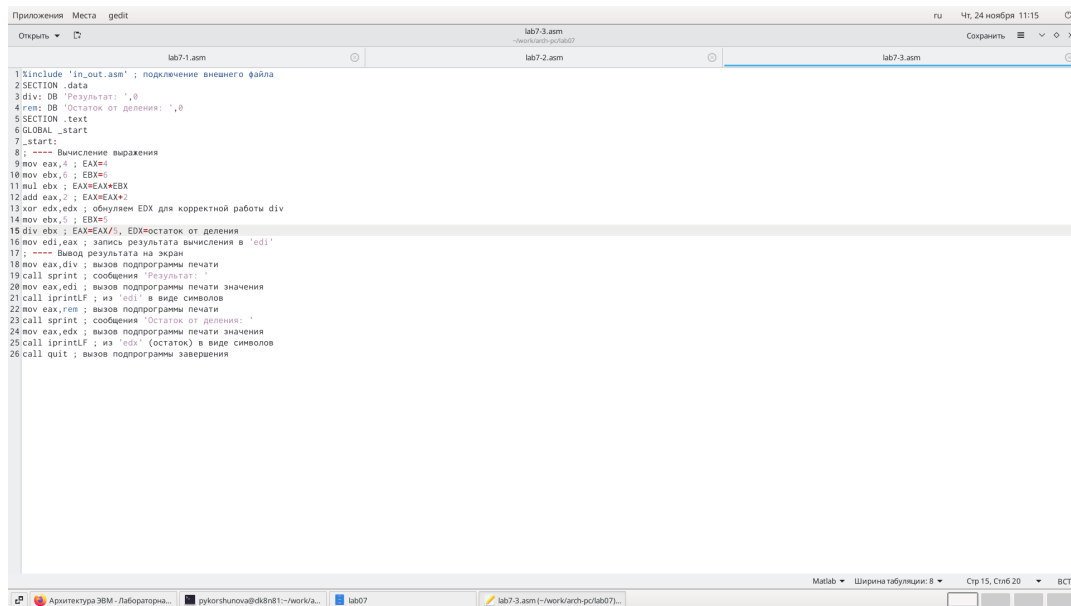


Рис. 2.11: Создаю файл lab7-3.asm

12. Проверяю результат (рис. 2.12)

```

pykorshunova@dk8n81 ~/work/arch-pc/lab07 $ touch ~/work/arch-pc/lab07/lab7-3.asm
pykorshunova@dk8n81 ~/work/arch-pc/lab07 $ nasm -f elf lab7-3.asm
pykorshunova@dk8n81 ~/work/arch-pc/lab07 $ ld -m elf_i386 -o lab7-3 lab7-3.o
pykorshunova@dk8n81 ~/work/arch-pc/lab07 $ ./lab7-3
Результат: 4
Остаток от деления: 1
pykorshunova@dk8n81 ~/work/arch-pc/lab07 $

```

Рис. 2.12: Проверяю результат

13. Изменяю текст программы для вычисления выражения $(4 \times 6 + 2)/5$.

Создаю исполняемый файл и проверяю его работу (рис. 2.13)

```

pykorshunova@dk8n81 ~/work/arch-pc/lab07 $ nasm -f elf lab7-3.asm
pykorshunova@dk8n81 ~/work/arch-pc/lab07 $ ld -m elf_i386 -o lab7-3 lab7-3.o
pykorshunova@dk8n81 ~/work/arch-pc/lab07 $ ./lab7-3
Результат: 5
Остаток от деления: 1
pykorshunova@dk8n81 ~/work/arch-pc/lab07 $

```

Рис. 2.13: Проверяю результат

14. Создаю файл `variant.asm` в каталоге `~/work/arch-pc/lab07`, внимательно изучаю текст программы из листинга 7.4 и ввожу в файл `variant.asm`. Проверяю результат (рис. 2.14)

```
pykorshunova@dk8n81 ~/work/arch-pc/lab07 $ touch ~/work/arch-pc/lab07/variant.asm
pykorshunova@dk8n81 ~/work/arch-pc/lab07 $ nasm -f elf variant.asm
pykorshunova@dk8n81 ~/work/arch-pc/lab07 $ ld -m elf_i386 -o variant variant.o
pykorshunova@dk8n81 ~/work/arch-pc/lab07 $ ./variant
Введите No студенческого билета:
1132226483
Ваш вариант: 4
pykorshunova@dk8n81 ~/work/arch-pc/lab07 $ 
```

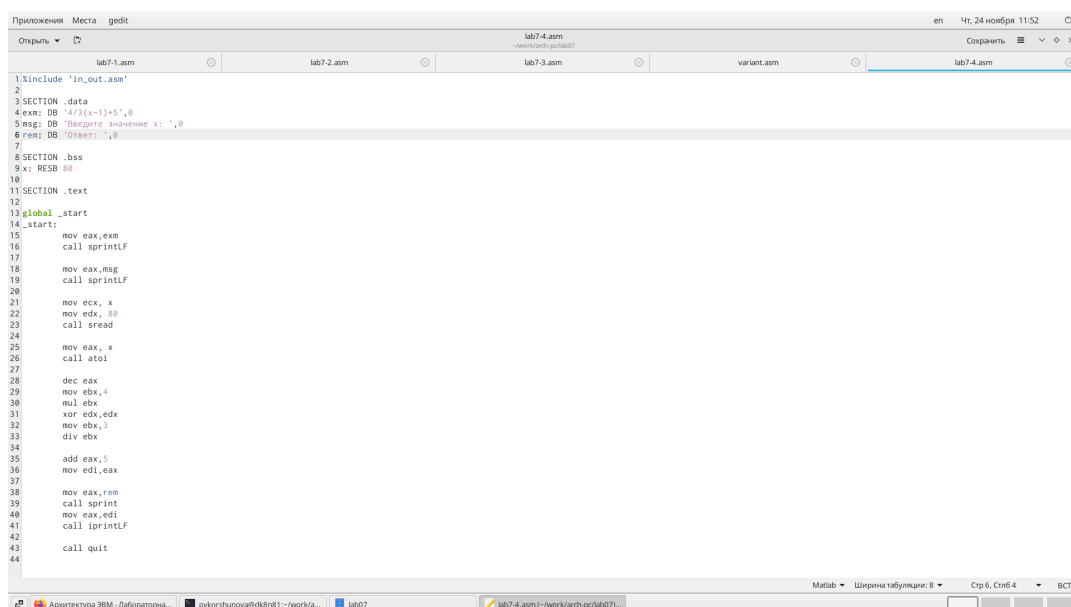
Рис. 2.14: Расчитываю вариант

3 Ответы на вопросы

1. `mov eax,msg call sprintLF`
2. Эти инструкции используются для ввода переменной X с клавиатуры и сохранения введенных данных
3. Эта инструкция используется для преобразования кода переменной ASCII в число
4. `mov ebx,20 div ebx inc edx`
5. В регистре ebx
6. Для увеличения значения edx на 1
7. `mov eax,edx call iprintLF`

4 Выполнение задания для самостоятельной работы

1. Пишу программу вычисления выражения по своему варианту (рис. 4.1)



```
1 %include "in_out.asm"
2
3 SECTION .data
4 evn: DB "4/3*(x-1)>5",0
5 msg: DB "Введите значение x: ",0
6 res: DB "Ответ: ",0
7
8 SECTION .bss
9 x: RESB 80
10
11 SECTION .text
12
13 global _start
14 _start:
15     mov eax, evn
16     call sprintf
17
18     mov eax, msg
19     call sprintf
20
21     mov ecx, x
22     mov edx, 80
23     call read
24
25     mov eax, x
26     call atoi
27
28     dec eax
29     mov ebx, 4
30     mul ebx
31     xor edx, edx
32     mov ebx, 3
33     div ebx
34
35     add eax, 5
36     mov edi, eax
37
38     mov eax, res
39     call sprintf
40     mov eax, edi
41     call sprintf
42
43     call quit
44
```

Рис. 4.1: Пишу программу вычисления выражения по своему варианту

2. Проверяю результат (рис. 4.2)

```
pykorshunova@dk8n81 ~/work/arch-pc/lab07 $ nasm -f elf lab7-4.asm
pykorshunova@dk8n81 ~/work/arch-pc/lab07 $ ld -m elf_i386 -o lab7-4 lab7-4.o
pykorshunova@dk8n81 ~/work/arch-pc/lab07 $ ./lab7-4
4/3(x-1)+5
Введите значение x:
4
Ответ: 9
pykorshunova@dk8n81 ~/work/arch-pc/lab07 $ nasm -f elf lab7-4.asm
pykorshunova@dk8n81 ~/work/arch-pc/lab07 $ ld -m elf_i386 -o lab7-4 lab7-4.o
pykorshunova@dk8n81 ~/work/arch-pc/lab07 $ ./lab7-4
4/3(x-1)+5
Введите значение x:
10
Ответ: 17
pykorshunova@dk8n81 ~/work/arch-pc/lab07 $ 
```

Рис. 4.2: Проверяю результат

5 Выводы

Я освоила арифметические инструкции языка ассемблера NASM.