

Отчёта по лабораторной работе

Дисциплина: Архитектура ЭВМ

Перегудов Александр Вадимович

Содержание

1	Цель работы	5
2	Задание	6
3	Теоретическое введение	7
4	Выполнение лабораторной работы	8
5	Выводы	20
	Список литературы	21

Список иллюстраций

4.1	Создание каталога и файла	8
4.2	Открытие gedit с помощью консоли	8
4.3	Текст программы	9
4.4	Результат работы программы	9
4.5	Результат работы программы	10
4.6	Создание файла с помощью команды touch	11
4.7	Текст программы	11
4.8	Результат работы программы	11
4.9	Изменённый текст программы	12
4.10	Результат работы программы	12
4.11	Изменённый текст программы	12
4.12	Результат работы программы	13
4.13	Создание файла с помощью команды touch	13
4.14	Текст программы	14
4.15	Результат работы программы	14
4.16	Изменённый текст программы	15
4.17	Результат работы программы	15
4.18	Создание файла с помощью команды touch	15
4.19	Проверка работоспособности	16
4.20	Результат работы программы	17
4.21	Создание файла с помощью команды touch	17
4.22	Текст программы	18
4.23	Трансляция и компоновка	18
4.24	Результат работы программы	18
4.25	Результат работы программы	19

Список таблиц

1 Цель работы

Освоение арифметических инструкций языка ассемблера NASM.

2 Задание

Здесь приводится описание задания в соответствии с рекомендациями методического пособия и выданным вариантом.

3 Теоретическое введение

4 Выполнение лабораторной работы

1. Создайте каталог для программ лабораторной работы № 6, перейшёл в него и создал файл lab6-1.asm. (рис. 4.1)

```
avperegudov@dk8n77 ~ $ mkdir ~/work/arch-pc/lab06
avperegudov@dk8n77 ~ $ cd ~/work/arch-pc/lab06
avperegudov@dk8n77 ~/work/arch-pc/lab06 $ touch lab6-1.asm
avperegudov@dk8n77 ~/work/arch-pc/lab06 $ ls
lab6-1.asm
```

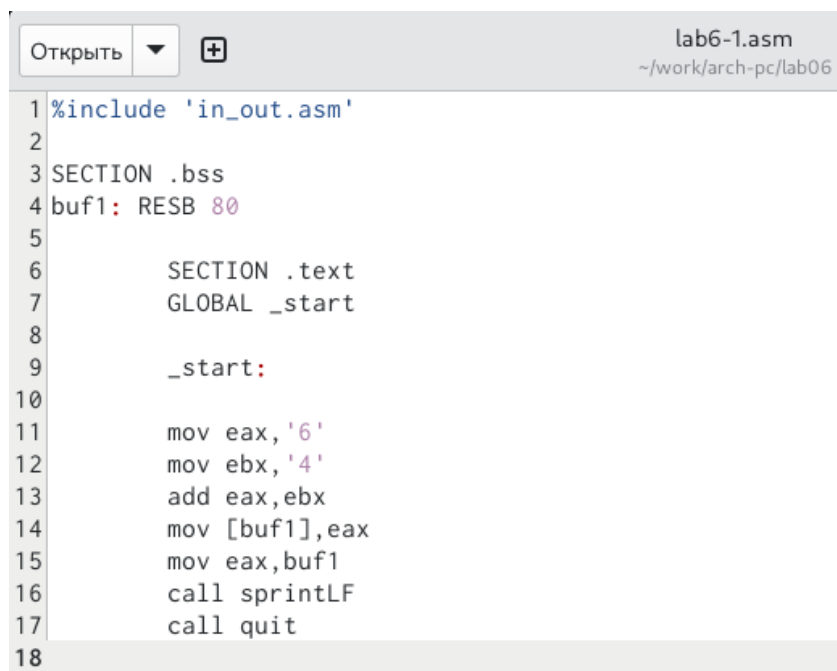
Рис. 4.1: Создание каталога и файла

2. Используя gedit для редактирование файла lab6-1.asm (рис. 4.2).

```
avperegudov@dk8n77 ~/work/arch-pc/lab06 $ gedit lab6-1.asm
```

Рис. 4.2: Открытие gedit с помощью консоли

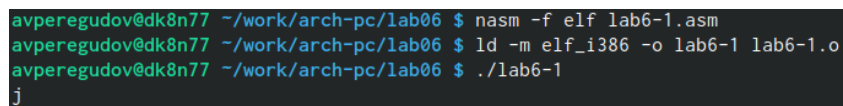
3. Ввел в файл lab6-1.asm текст программы из листинга 6.1 (рис. 4.3).



```
1 %include 'in_out.asm'
2
3 SECTION .bss
4 buf1: RESB 80
5
6     SECTION .text
7     GLOBAL _start
8
9     _start:
10
11     mov eax, '6'
12     mov ebx, '4'
13     add eax, ebx
14     mov [buf1], eax
15     mov eax, buf1
16     call sprintLF
17     call quit
18
```

Рис. 4.3: Текст программы

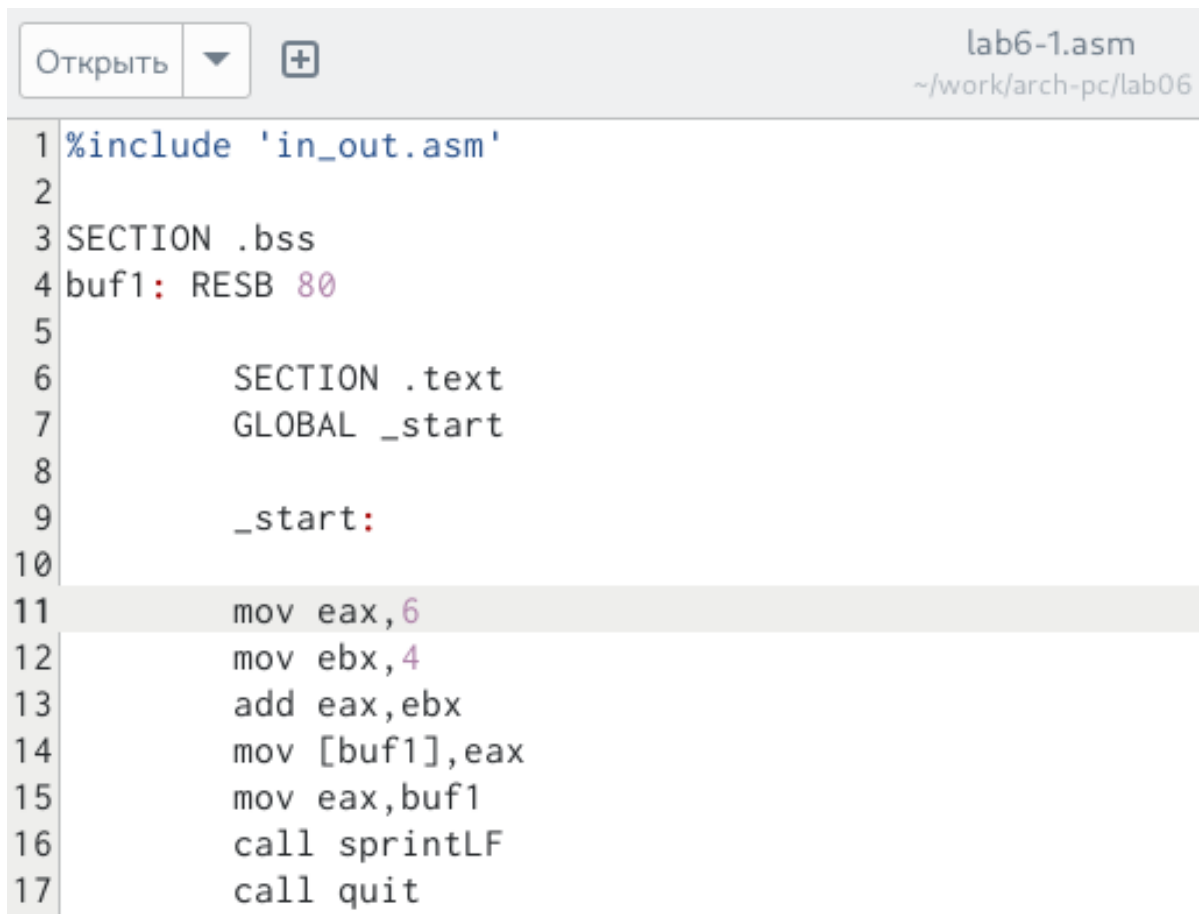
4. Создал исполняемый файл и запустил его. (рис. 4.4)



```
avperegudov@dk8n77 ~/work/arch-pc/lab06 $ nasm -f elf lab6-1.asm
avperegudov@dk8n77 ~/work/arch-pc/lab06 $ ld -m elf_i386 -o lab6-1 lab6-1.o
avperegudov@dk8n77 ~/work/arch-pc/lab06 $ ./lab6-1
j
```

Рис. 4.4: Результат работы программы

5. Изменил текст программы и вместо символов, записал в регистры числа.
(рис. ??)

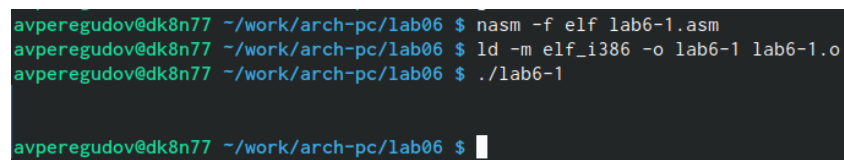


```
lab6-1.asm
~/work/arch-pc/lab06

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

{#fig:005 width=70%}

6. Создал исполняемый файл и запустил его. {4.5 width=70%}



```
avperegudov@dk8n77 ~/work/arch-pc/lab06 $ nasm -f elf lab6-1.asm
avperegudov@dk8n77 ~/work/arch-pc/lab06 $ ld -m elf_i386 -o lab6-1 lab6-1.o
avperegudov@dk8n77 ~/work/arch-pc/lab06 $ ./lab6-1

avperegudov@dk8n77 ~/work/arch-pc/lab06 $
```

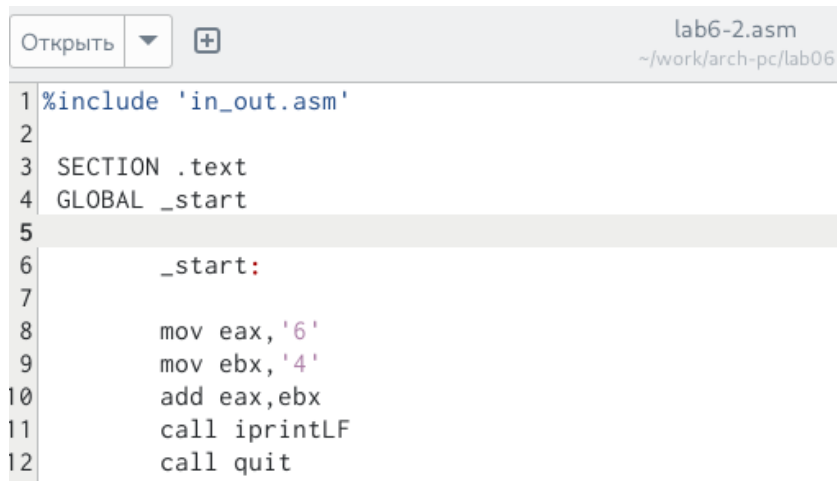
Рис. 4.5: Результат работы программы

7. Создал файл lab6-2.asm в каталоге ~/work/arch-pc/lab06 {4.6 width=70%}.

```
avperegudov@dk8n77 ~/work/arch-pc/lab06 $ touch lab6-2.asm
avperegudov@dk8n77 ~/work/arch-pc/lab06 $ ls
in_out.asm lab6-1 lab6-1.asm lab6-1.o lab6-2.asm
avperegudov@dk8n77 ~/work/arch-pc/lab06 $
```

Рис. 4.6: Создание файла с помощью команды touch

8. Ввел в файл lab6-2.asm текст программы из листинга 6.2 {?? width=70%}.



```
lab6-2.asm
~/work/arch-pc/lab06

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

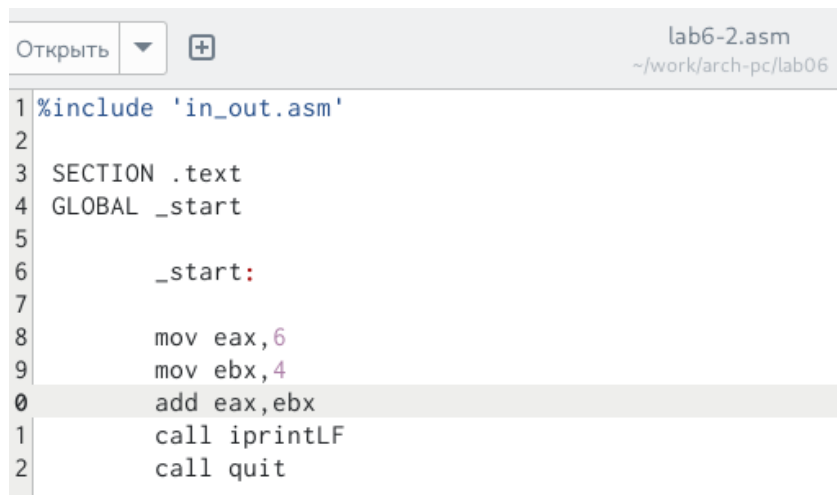
Рис. 4.7: Текст программы

9. Создал исполняемый файл и запустил его.{рис. 4.8}

```
avperegudov@dk8n77 ~/work/arch-pc/lab06 $ nasm -f elf lab6-2.asm
avperegudov@dk8n77 ~/work/arch-pc/lab06 $ ld -m elf_i386 -o lab6-2 lab6-2.o
avperegudov@dk8n77 ~/work/arch-pc/lab06 $ ./lab6-2
106
```

Рис. 4.8: Результат работы программы

10. Изменил текст программы и вместо символов, записал в регистры числа.(рис. 4.9).

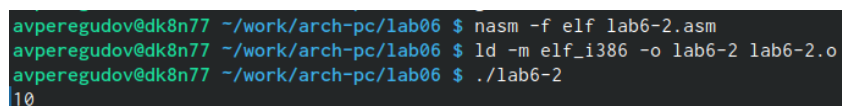


```
lab6-2.asm
~/work/arch-pc/lab06

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

Рис. 4.9: Изменённый текст программы

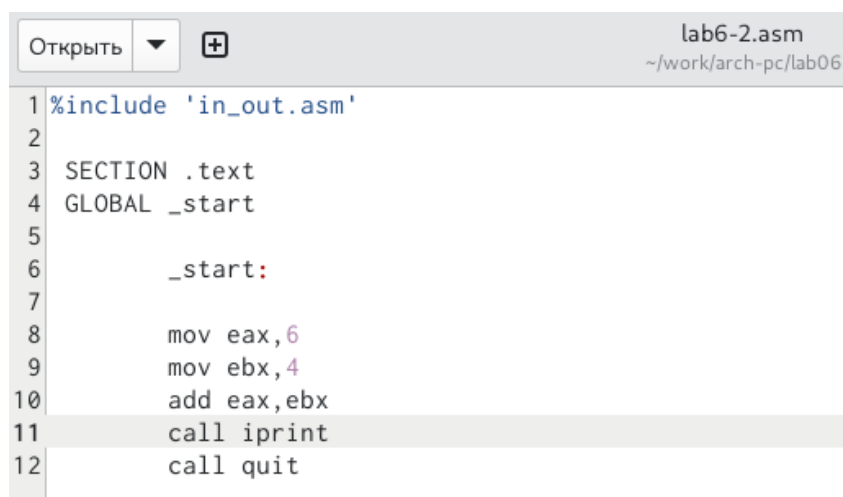
11. Создал исполняемый файл и запустил его.{4.10 width=70%}



```
avperegudov@dk8n77 ~/work/arch-pc/lab06 $ nasm -f elf lab6-2.asm
avperegudov@dk8n77 ~/work/arch-pc/lab06 $ ld -m elf_i386 -o lab6-2 lab6-2.o
avperegudov@dk8n77 ~/work/arch-pc/lab06 $ ./lab6-2
10
```

Рис. 4.10: Результат работы программы

12. Заменял функцию iprintLF на iprint.{4.11 width=70%}



```
lab6-2.asm
~/work/arch-pc/lab06

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

Рис. 4.11: Изменённый текст программы

13. Создал исполняемый файл и запустил его.{4.12 width=70%}

```
avperegudov@dk8n77 ~/work/arch-pc/lab06 $ nasm -f elf lab6-2.asm
avperegudov@dk8n77 ~/work/arch-pc/lab06 $ ld -m elf_i386 -o lab6-2 lab6-2.o
avperegudov@dk8n77 ~/work/arch-pc/lab06 $ ./lab6-2
10avperegudov@dk8n77 ~/work/arch-pc/lab06 $
```

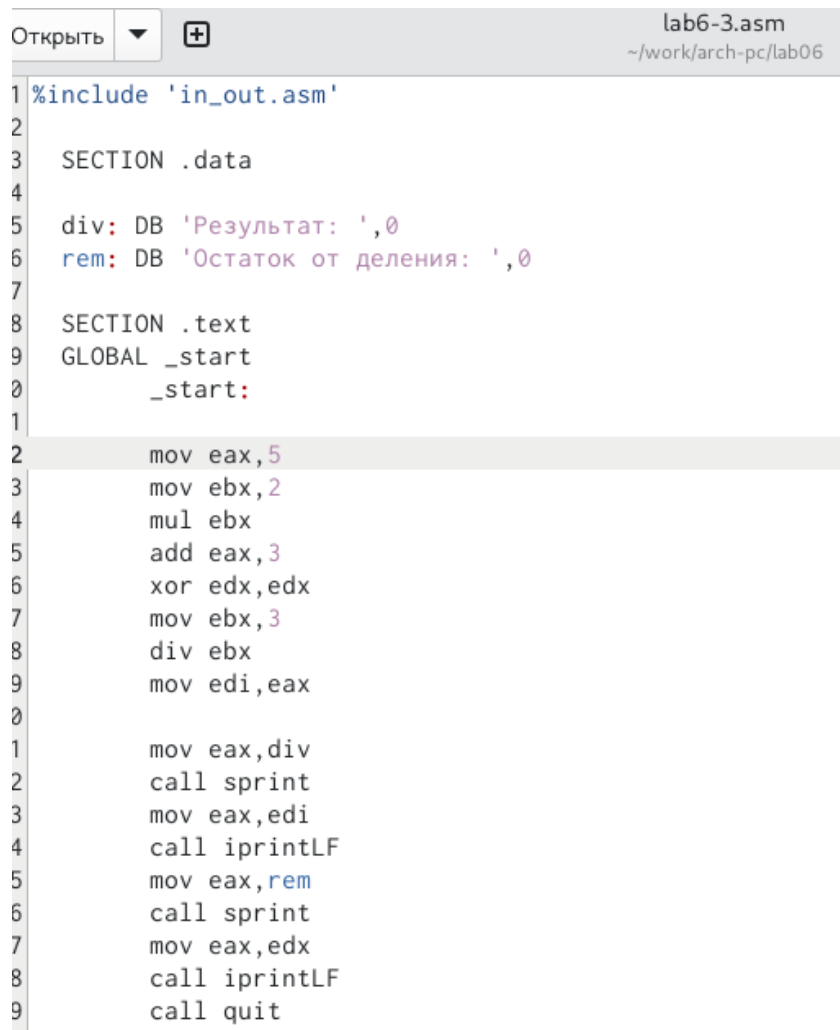
Рис. 4.12: Результат работы программы

14. Создайте файл lab6-3.asm в каталоге ~/work/arch-pc/lab06.{4.13 width=70%}

```
avperegudov@dk8n77 ~/work/arch-pc/lab06 $ touch lab6-3.asm
avperegudov@dk8n77 ~/work/arch-pc/lab06 $ ls
in_out.asm  lab6-1.asm  lab6-2      lab6-2.o  lab6-3.asm  variant    variant.o
lab6-1      lab6-1.o    lab6-2.asm  lab6-3    lab6-3.o    variant.asm
avperegudov@dk8n77 ~/work/arch-pc/lab06 $
```

Рис. 4.13: Создание файла с помощью команды touch

15. Ввел в файл lab6-3.asm текст программы из листинга 6.3.{4.14 width=70%}

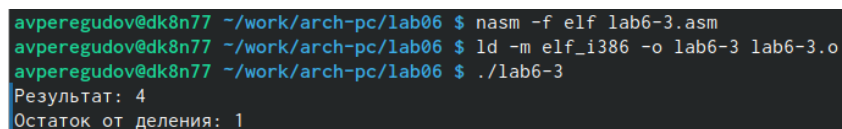


```
Открыть ▼ + lab6-3.asm
~/work/arch-pc/lab06

1 %include 'in_out.asm'
2
3 SECTION .data
4
5 div: DB 'Результат: ',0
6 rem: DB 'Остаток от деления: ',0
7
8 SECTION .text
9 GLOBAL _start
10 _start:
11
12 mov eax,5
13 mov ebx,2
14 mul ebx
15 add eax,3
16 xor edx,edx
17 mov ebx,3
18 div ebx
19 mov edi,eax
20
21 mov eax,div
22 call sprint
23 mov eax,edi
24 call iprintLF
25 mov eax,rem
26 call sprint
27 mov eax,edx
28 call iprintLF
29 call quit
```

Рис. 4.14: Текст программы

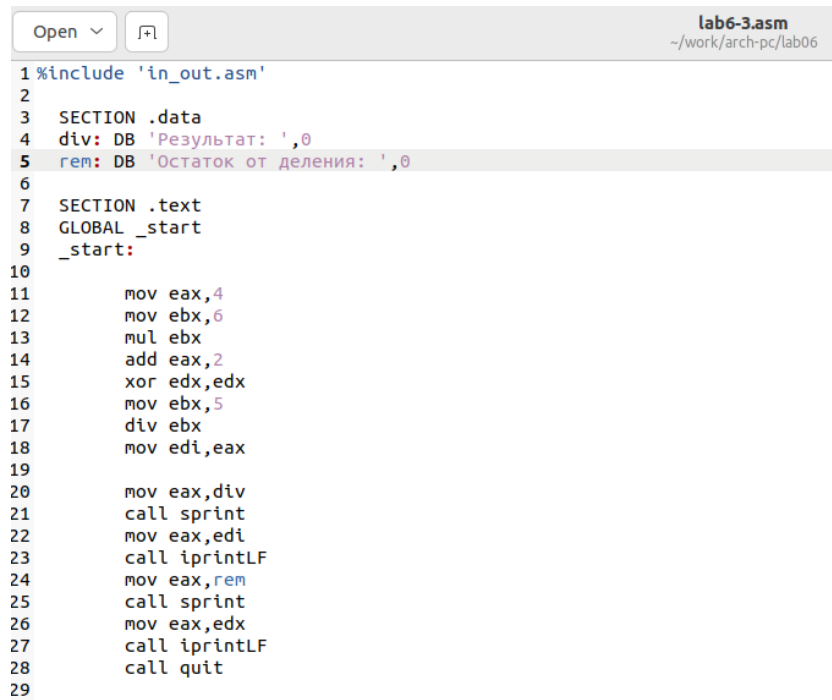
16. Создал исполняемый файл и запустил его.{4.15 width=70%}



```
avperegudov@dk8n77 ~/work/arch-pc/lab06 $ nasm -f elf lab6-3.asm
avperegudov@dk8n77 ~/work/arch-pc/lab06 $ ld -m elf_i386 -o lab6-3 lab6-3.o
avperegudov@dk8n77 ~/work/arch-pc/lab06 $ ./lab6-3
Результат: 4
Остаток от деления: 1
```

Рис. 4.15: Результат работы программы

17. Изменил текст программы для вычисления результата другого выражения.{4.16 width=70%}



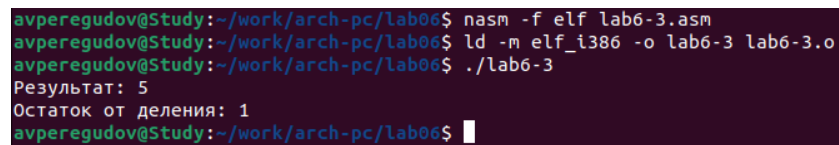
```

1 %include 'in_out.asm'
2
3 SECTION .data
4 div: DB 'Результат: ',0
5 rem: DB 'Остаток от деления: ',0
6
7 SECTION .text
8 GLOBAL _start
9 _start:
10
11     mov eax,4
12     mov ebx,6
13     mul ebx
14     add eax,2
15     xor edx,edx
16     mov ebx,5
17     div ebx
18     mov edi,eax
19
20     mov eax,div
21     call sprint
22     mov eax,edi
23     call iprintLF
24     mov eax,rem
25     call sprint
26     mov eax,edx
27     call iprintLF
28     call quit
29

```

Рис. 4.16: Изменённый текст программы

18. Создал исполняемый файл и запустил его.{4.17 width=70%}



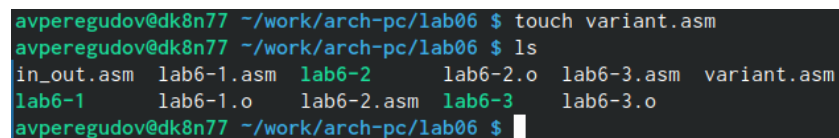
```

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

```

Рис. 4.17: Результат работы программы

19. Создал файл variant.asm в каталоге ~/work/arch-pc/lab06.{4.18 width=70%}



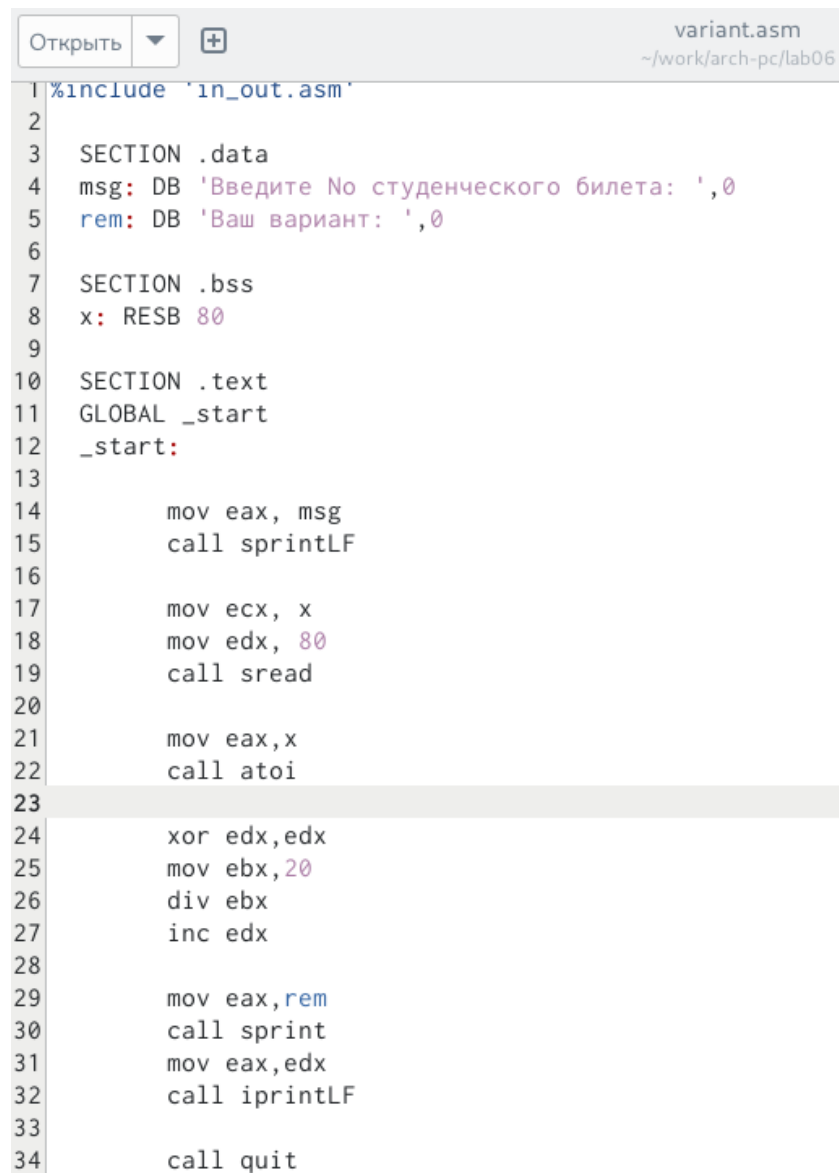
```

avperegudov@dk8n77 ~/work/arch-pc/lab06 $ touch variant.asm
avperegudov@dk8n77 ~/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
avperegudov@dk8n77 ~/work/arch-pc/lab06 $

```

Рис. 4.18: Создание файла с помощью команды touch

20. Ввел в файл variant.asm текст программы из листинга 6.4.{4.19 width=70%}



```
1 %include 'in_out.asm'
2
3 SECTION .data
4 msg: DB 'Введите No студенческого билета: ',0
5 rem: DB 'Ваш вариант: ',0
6
7 SECTION .bss
8 x: RESB 80
9
10 SECTION .text
11 GLOBAL _start
12 _start:
13
14     mov eax, msg
15     call sprintf
16
17     mov ecx, x
18     mov edx, 80
19     call sread
20
21     mov eax, x
22     call atoi
23
24     xor edx, edx
25     mov ebx, 20
26     div ebx
27     inc edx
28
29     mov eax, rem
30     call sprintf
31     mov eax, edx
32     call iprintLF
33
34     call quit
```

Рис. 4.19: Проверка работоспособности

21. Создал исполняемый файл, запустил его и ввёл номер своего студенческого билета.{4.20 width=70%}


```

avperegudov@dk8n77 ~/work/arch-pc/lab06 $ nasm -f elf variant.asm
avperegudov@dk8n77 ~/work/arch-pc/lab06 $ ld -m elf_i386 -o variant variant.o
avperegudov@dk8n77 ~/work/arch-pc/lab06 $ ./variant
Введите No студенческого билета:
1132239659
Ваш вариант: 20
avperegudov@dk8n77 ~/work/arch-pc/lab06 $

```

Рис. 4.20: Результат работы программы

22. Создал файл lab6-4.asm в каталоге ~/work/arch-pc/lab06.{4.21 width=70%}

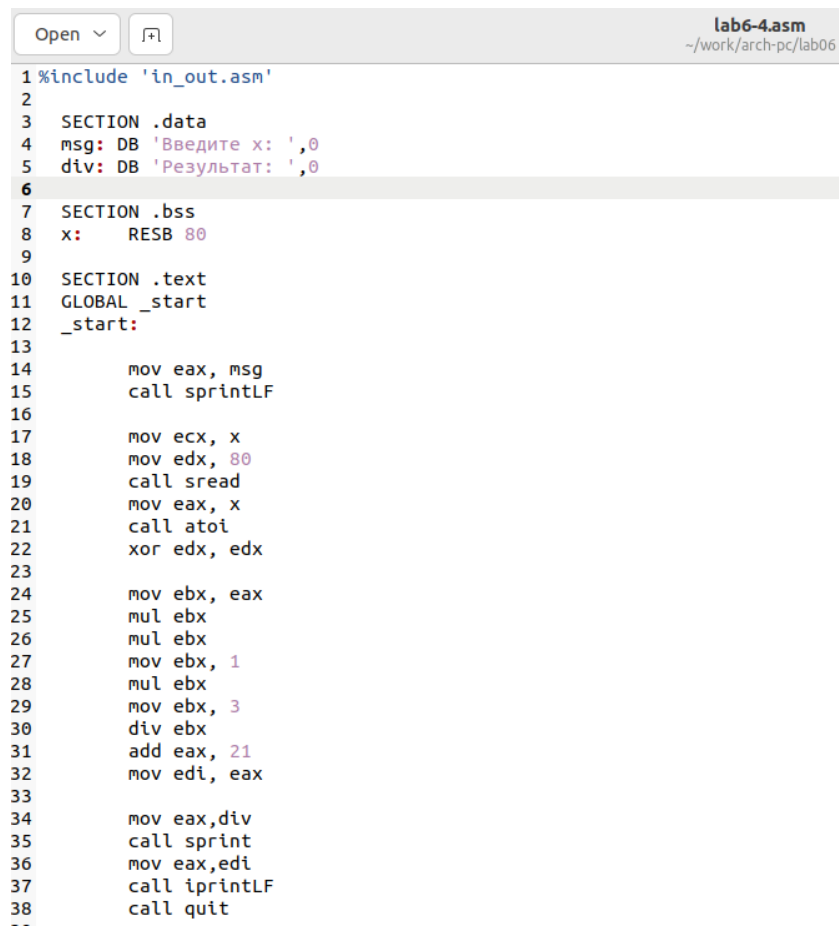
```

avperegudov@Study:~/work/arch-pc/lab06$ touch lab6-4.asm
avperegudov@Study:~/work/arch-pc/lab06$

```

Рис. 4.21: Создание файла с помощью команды touch

22. Ввёл код для вычисления значения функции от x из таблицы 6.3 под номером 20.{4.22 width=70%}



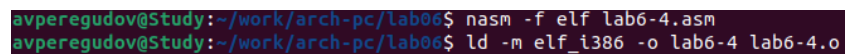
```

1 %include 'in_out.asm'
2
3 SECTION .data
4 msg: DB 'Введите x: ',0
5 div: DB 'Результат: ',0
6
7 SECTION .bss
8 x: RESB 80
9
10 SECTION .text
11 GLOBAL _start
12 _start:
13
14     mov eax, msg
15     call sprintf
16
17     mov ecx, x
18     mov edx, 80
19     call sread
20     mov eax, x
21     call atoi
22     xor edx, edx
23
24     mov ebx, eax
25     mul ebx
26     mul ebx
27     mov ebx, 1
28     mul ebx
29     mov ebx, 3
30     div ebx
31     add eax, 21
32     mov edi, eax
33
34     mov eax, div
35     call sprintf
36     mov eax, edi
37     call iprintf
38     call quit
--

```

Рис. 4.22: Текст программы

23. Создал исполняемый файл.{4.23 width=70%}



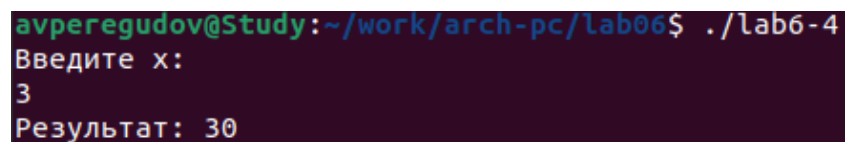
```

avperegudov@Study:~/work/arch-pc/lab06$ nasm -f elf lab6-4.asm
avperegudov@Study:~/work/arch-pc/lab06$ ld -m elf_i386 -o lab6-4 lab6-4.o

```

Рис. 4.23: Трансляция и компоновка

24. Запустил исполняемый файл и ввёл первое значение x.{4.24 width=70%}



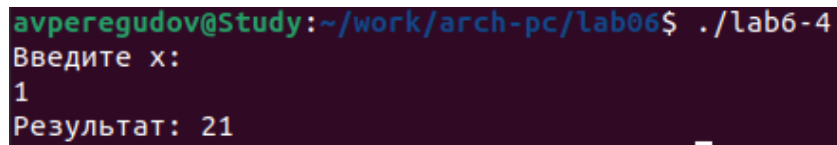
```

avperegudov@Study:~/work/arch-pc/lab06$ ./lab6-4
Введите x:
3
Результат: 30

```

Рис. 4.24: Результат работы программы

25. Запустил исполняемый файл и ввёл второе значение x.{4.25 width=70%}



```
avperegudov@Study:~/work/arch-pc/lab06$ ./lab6-4
Введите x:
1
Результат: 21
```

Рис. 4.25: Результат работы программы

#Ответы на вопросы

#1.

mov eax, rem call sprint

#2. Для считывания сообщения.

#3. Для преобразования ascii-код символа в целое число.

#4.

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

#5. edx.

#6. Для прибавления к единицы к значению лежащему в edx.

#7.

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

5 Выводы

В этой работе были освоены арифметические инструкции языка ассемблера NASM.

Список литературы