

Отчёт по лабораторной работе №7

Тарутина Кристина Олеговна

Содержание

1	Цель работы	1
2	Выполнение лабораторной работы.....	1
3	Выполнение самостоятельной работы.....	7
4	Выводы	8

1 Цель работы

Освоение арифметических инструкций языка ассемблера NASM

2 Выполнение лабораторной работы

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



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

Рис. 1: Создаю файл

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



```
*lab7-1.asm
~fwork/arch-pc/lab07

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

Рис. 2: Текст программы

Создаю исполняемый файл и запускаю его.(рис. 3)

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

Рис. 3: Создание исполняемого файла и работа программы

Изменяю текст программы (рис. 4)

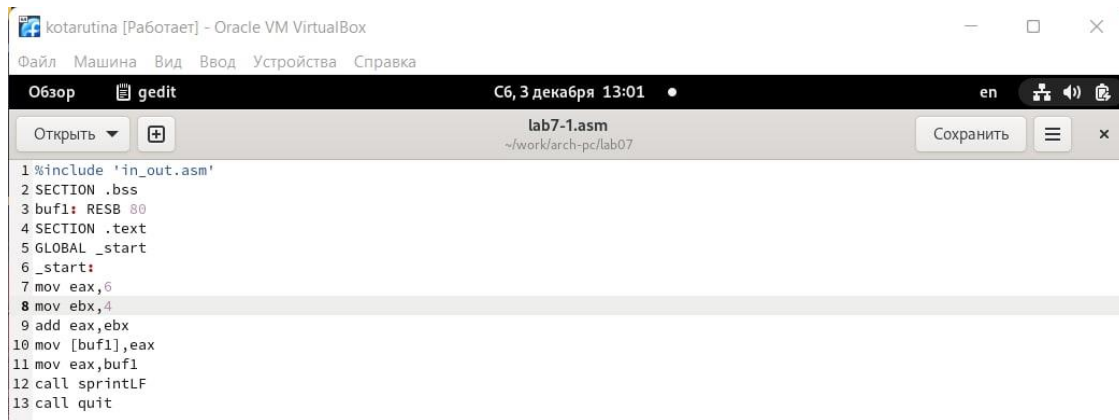


Рис. 4: Изменённый текст программы

Создаю исполняемый файл и проверяю его работу(рис. 5) Да, при выводе на экран символ отображается, так как символ с этим кодом это символ переноса строки, то и видим мы перенос строки

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

[kotarutina@fedora lab07]$
```

Рис. 5: Создание исполняемого файла и работа программы

Создаю файл lab7-2.asm(рис. 6)

```
[kotarutina@fedora lab07]$ touch ~/work/arch-pc/lab07/lab7-2.asm
[kotarutina@fedora lab07]$
```

Рис. 6: Создание файла

Ввожу в него код из Листинга 7.2(рис. 7)



Рис. 7: Текст программы

Создаю исполняемый файл и запускаю его(рис. 8)

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

Рис. 8: Создание исполняемого файла и работа программы

Изменяю код программы(рис. 9)



Рис. 9: Изменённый код

Создаю исполняемый файл и проверяю его работу. В результате мы получим число 10 при исполнении команды(рис. 10)

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

Рис. 10: Работа программы

В тексте программы заменяю функцию iprintLF на iprint(рис. 11)



Рис. 11: Изменённый код

Создаю исполняемый файл и запускаю его. При смене функции после результата программа не вывела символ переноса строки(рис. 12)

```

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

```

Рис. 12: Работа программы

Создаю файл lab7-3.asm в каталоге ~/work/arch-pc/lab07 Внимательно изучаю текст программы из листинга 7.3 и ввожу в lab7- 3.asm. (рис. 13)

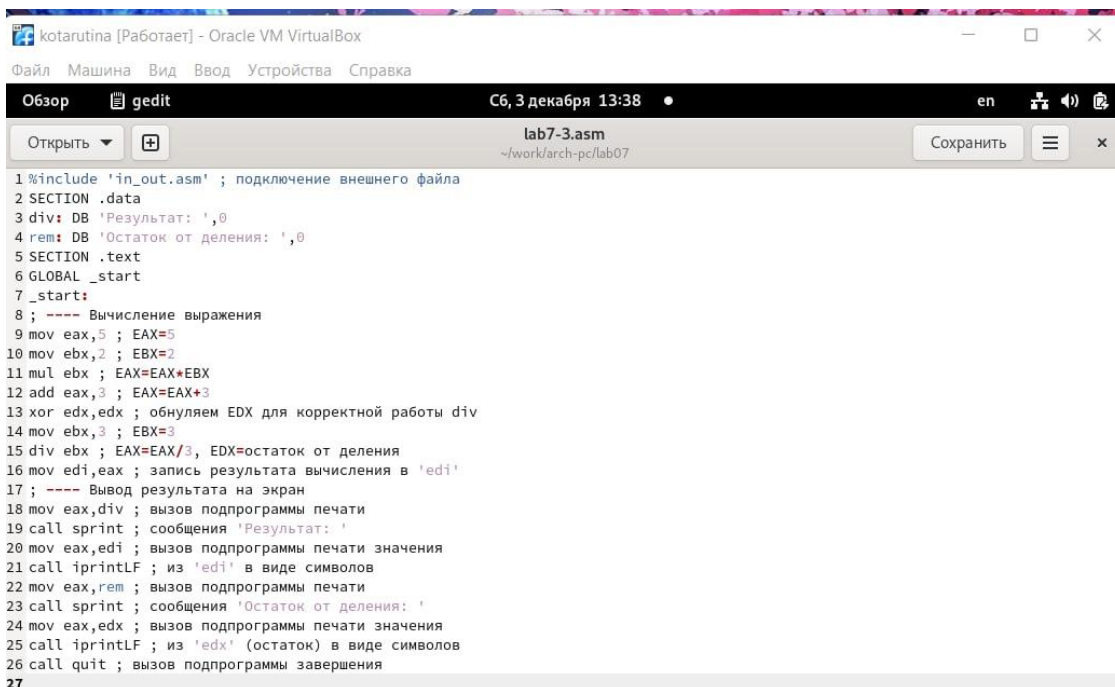


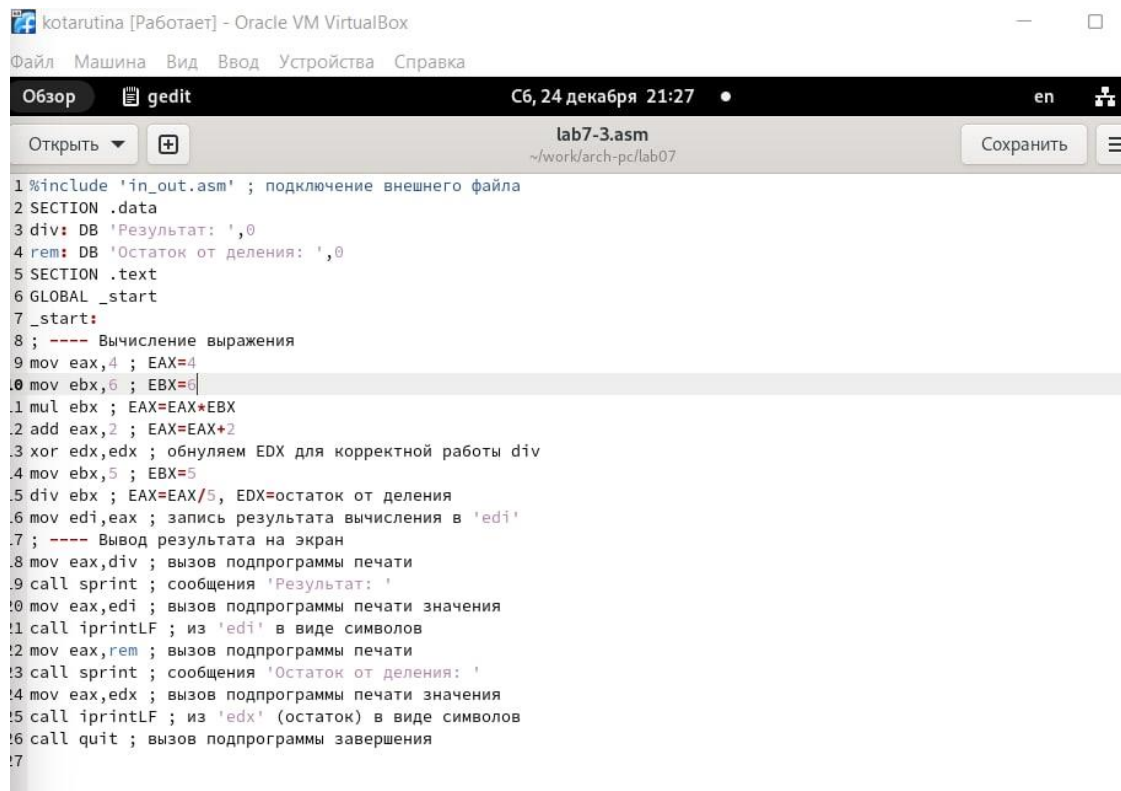
Рис. 13: Код

Создаю исполняемый файл и запускаю его.(рис. 14)

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

Рис. 14: Работа файла

Изменяю текст программы для вычисления выражения $f(x) = (4 * 6 + 2)/5$ (рис. 15)



```
1 %include 'in_out.asm' ; подключение внешнего файла
2 SECTION .data
3 div: DB 'Результат: ',0
4 rem: DB 'Остаток от деления: ',0
5 SECTION .text
6 GLOBAL _start
7 _start:
8 ; ---- Вычисление выражения
9 mov eax,4 ; EAX=4
10 mov ebx,6 ; EBX=6
11 mul ebx ; EAX=EAX*EBX
12 add eax,2 ; EAX=EAX+2
13 xor edx,edx ; обнуляем EDX для корректной работы div
14 mov ebx,5 ; EBX=5
15 div ebx ; EAX=EAX/5, EDX=остаток от деления
16 mov edi,eax ; запись результата вычисления в 'edi'
17 ; ---- Вывод результата на экран
18 mov eax,div ; вызов подпрограммы печати
19 call sprint ; сообщения 'Результат: '
20 mov eax,edi ; вызов подпрограммы печати значения
21 call iprintLF ; из 'edi' в виде символов
22 mov eax,rem ; вызов подпрограммы печати
23 call sprint ; сообщения 'Остаток от деления: '
24 mov eax,edx ; вызов подпрограммы печати значения
25 call iprintLF ; из 'edx' (остаток) в виде символов
26 call quit ; вызов подпрограммы завершения
27
```

Рис. 15: Изменённый текст программы

Создаю исполняемый файл и проверяю его работу. (рис. 16)

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

Рис. 16: Работа программы

Создаю файл variant.asm в каталоге ~/work/arch-pc/lab07 (рис. 17)

```
[kotarutina@fedora lab07]$ touch ~/work/arch-pc/lab07/variant.asm
[kotarutina@fedora lab07]$
```

Рис. 17: Создание файла

Внимательно изучаю текст программы из листинга 7.4 и ввожу в файл variant.asm. (рис. 18)



```
1 %include 'in_out.asm'
2 SECTION .data
3 msg: DB 'Введите № студенческого билета: ',0
4 rem: DB 'Ваш вариант: ',0
5 SECTION .bss
6 x: RESB 80
7 SECTION .text
8 GLOBAL _start
9 _start:
10 mov eax, msg
11 call sprintf
12 mov ecx, x
13 mov edx, 80
14 call sread
15 mov eax,x ; вызов подпрограммы преобразования
16 call atoi ; ASCII кода в число, 'eax=x'
17 xor edx,edx
18 mov ebx,20
19 div ebx
20 inc edx
21 mov eax,rem
22 call sprint
23 mov eax,edx
24 call iprintLF
25 call quit
```

Рис. 18: текст программы

Создаю исполняемый файл и запускаю его.(рис. 19) Аналитически проверю. Программа корректна

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? nasm mov ecx, x mov edx, 80 call sread

Для считывания номера студ. билета

3. Для чего используется инструкция "call atoi"?

Для преобразования ASCII кода в число

4. Какие строки листинга 7.4 отвечают за вычисления варианта?

mov ebx,20

div ebx

inc edx

5. В какой регистр записывается остаток от деления при выполнении инструкции “div ebx”?

edx

6. Для чего используется инструкция “inc edx”?

Для прибавления единицы к остатку

7. Какие строки листинга 7.4 отвечают за вывод на экран результата вычислений?

mov eax,edx

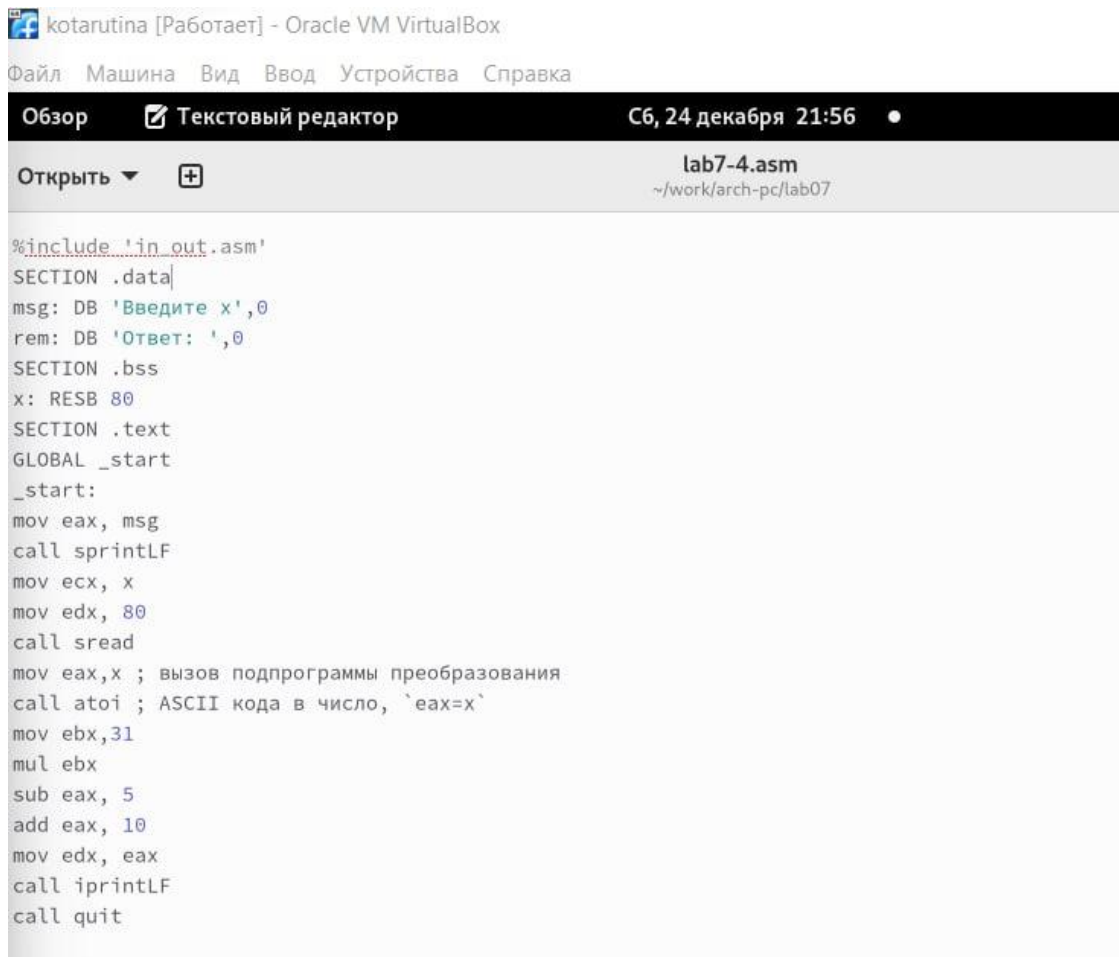
call iprintLF

```
[kotarutina@fedora lab07]$ nasm -f elf variant.asm
[kotarutina@fedora lab07]$ ld -m elf_i386 -o variant variant.o
[kotarutina@fedora lab07]$ ./variant
Введите № студенческого билета:
1132226528
Ваш вариант: 9
```

Рис. 19: Работа программы

3 Выполнение самостоятельной работы

Пишу программу вычисления выражения $y = f(x)$, $y = 10 + (31x - 5)$ (вариант 9). (рис. 20)

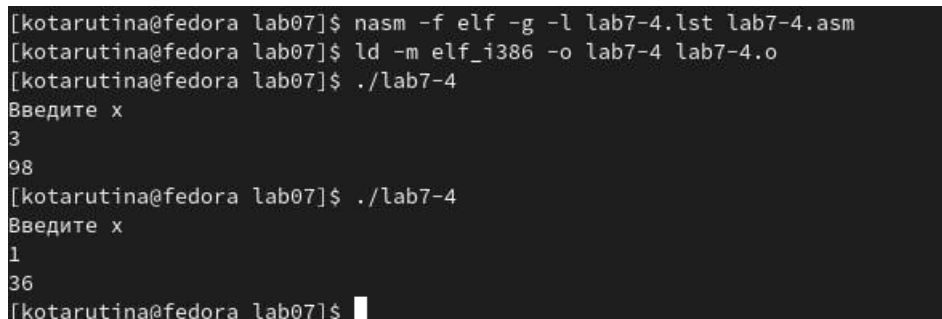


The screenshot shows a text editor window titled "kotarutina [Работает] - Oracle VM VirtualBox". The menu bar includes "Файл", "Машина", "Вид", "Ввод", "Устройства", and "Справка". The toolbar has "Обзор", "Текстовый редактор", and a date/time display "С6, 24 декабря 21:56". The file name is "lab7-4.asm" located at "~/.work/arch-pc/lab07". The code content is as follows:

```
%include 'in_out.asm'
SECTION .data
msg: DB 'Введите x',0
rem: DB 'Ответ: ',0
SECTION .bss
x: RESB 80
SECTION .text
GLOBAL _start
_start:
mov eax, msg
call sprintf
mov ecx, x
mov edx, 80
call sread
mov eax, x ; вызов подпрограммы преобразования
call atoi ; ASCII кода в число, `eax=x`
mov ebx, 31
mul ebx
sub eax, 5
add eax, 10
mov edx, eax
call iprintLF
call quit
```

Рис. 20: Текст программы

Создаю исполняемый файл и проверяю его работу для значений x_1 и x_2 (рис. 21)



The screenshot shows a terminal window with the following commands and output:

```
[kotarutina@fedora lab07]$ nasm -f elf -g -l lab7-4.lst lab7-4.asm
[kotarutina@fedora lab07]$ ld -m elf_i386 -o lab7-4 lab7-4.o
[kotarutina@fedora lab07]$ ./lab7-4
Введите x
3
98
[kotarutina@fedora lab07]$ ./lab7-4
Введите x
1
36
[kotarutina@fedora lab07]$
```

Рис. 21: Работа программы

4 Выводы

Освоение арифметических инструкций языка ассемблера NASM прошло успешно