

Лабораторная работа №7.

Арифметические операции в NASM

Боровиков Даниил Александрович

Содержание

1	Цель работы	4
2	Выполнение лабораторной работы	5
3	Ответы на вопросы	13
4	Самостоятельная работа	15
5	Выводы	17

Список иллюстраций

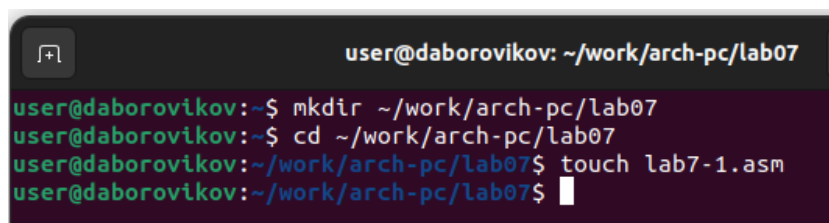
2.1	Создание файла lab7-1.asm в соответствующем каталоге	5
2.2	Текст программы из листинга 7.1.	5
2.3	Запуск исполняемого файла	6
2.4	Текст измененной программы	6
2.5	Запуск исправленного исполняемого файла	6
2.6	Листинг программы lab7-2.asm	7
2.7	Запуск исполняемого файла	7
2.8	Замена символов на числа	8
2.9	Запуск измененного исполняемого файла	8
2.10	Замена функции iprintLF на iprint	9
2.11	Запуск файла с iprint вместо iprintLF	9
2.12	Создание файла lab7-3.asm	9
2.13	Текст программы lab7-3.asm	10
2.14	Запуск исполняемого файла lab7-3.asm	10
2.15	Текст программы для вычисления выражения $\boxtimes(\boxtimes) = (4 \boxtimes 6 + 2)/5$	11
2.16	Запуск исполняемого файла для вычисления выражения $\boxtimes(\boxtimes) = (4 \boxtimes 6 + 2)/5$	11
2.17	Листинг программы variant.asm	12
2.18	Запуск исполняемого файла для нахождения номера варианта	12
4.1	Листинг программы sam.asm	15
4.2	Запуск исполняемого файла для решения выражения $5(x-1)^2$	16

1 Цель работы

Освоение арифметических инструкций языка ассемблера NASM.

2 Выполнение лабораторной работы

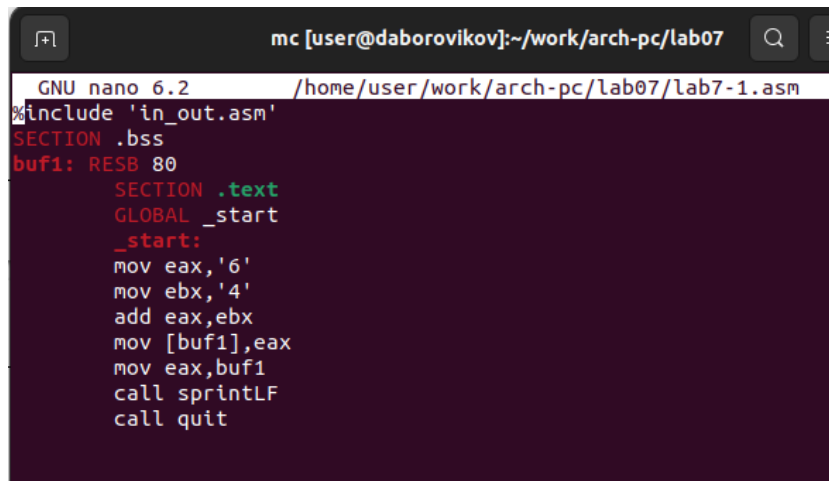
Создадим каталог для программ лабораторной работы № 7, перейдем в него и создадим файл lab7-1.asm(рис. 2.1)



```
user@daborovikov: ~/work/arch-pc/lab07
user@daborovikov:~$ mkdir ~/work/arch-pc/lab07
user@daborovikov:~$ cd ~/work/arch-pc/lab07
user@daborovikov:~/work/arch-pc/lab07$ touch lab7-1.asm
user@daborovikov:~/work/arch-pc/lab07$
```

Рис. 2.1: Создание файла lab7-1.asm в соответствующем каталоге

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



```
mc [user@daborovikov]:~/work/arch-pc/lab07
GNU nano 6.2 /home/user/work/arch-pc/lab07/lab7-1.asm
%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 sprintf
    call quit
```

Рис. 2.2: Текст программы из листинга 7.1.

Создадим исполняемый файл и запустим его.(рис. 2.3)

```

user@daborovikov:~/work/arch-pc/lab07$ nasm -f elf lab7-1.asm
user@daborovikov:~/work/arch-pc/lab07$ ld -m elf_i386 -o lab7-1 lab7-1.o
user@daborovikov:~/work/arch-pc/lab07$ ./lab7-1
j
user@daborovikov:~/work/arch-pc/lab07$

```

Рис. 2.3: Запуск исполняемого файла

Далее изменим текст программы и вместо символов, запишем в регистры числа.(рис. 2.4)

```

GNU nano 6.2 /home/user/work/arch-pc/lab07/lab7-
%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. 2.4: Текст измененной программы

Создадим исполняемый файл исправленного текста программы lab7-1.asm и запустите его.(рис. 2.5)

```

user@daborovikov:~/work/arch-pc/lab07$ nasm -f elf lab7-1.asm
user@daborovikov:~/work/arch-pc/lab07$ ld -m elf_i386 -o lab7-1 lab7-1.o
user@daborovikov:~/work/arch-pc/lab07$ ./lab7-1

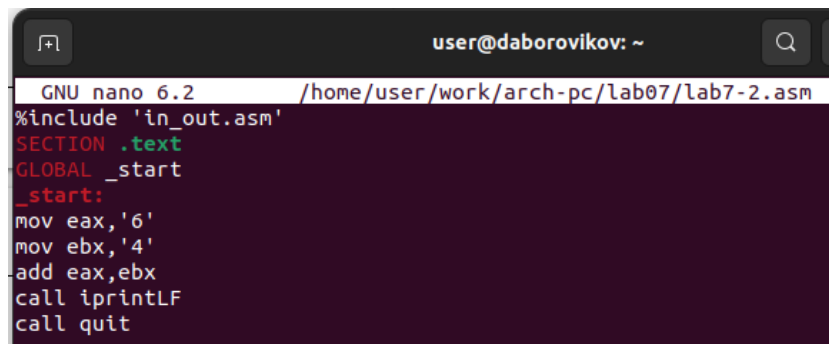
user@daborovikov:~/work/arch-pc/lab07$ mc

```

Рис. 2.5: Запуск исправленного исполняемого файла

Пользуясь таблицей ASCII определим, что код 10 соответствует символ LF, т.е. Данный символ не отображается на экране.

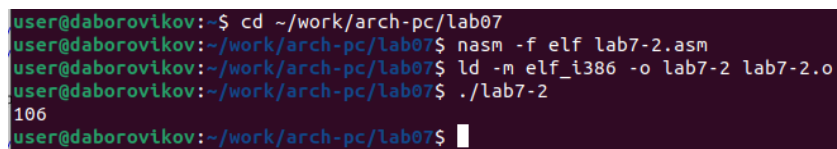
Создадим файл lab7-2.asm в каталоге ~/work/arch-pc/lab07 и введем в него текст программы из листинга 7.2.(рис. 2.6)



```
GNU nano 6.2 /home/user/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
```

Рис. 2.6: Листинг программы lab7-2.asm

Создадим исполняемый файл и запустим его(рис. 2.7)



```
user@daborovikov:~$ cd ~/work/arch-pc/lab07
user@daborovikov:~/work/arch-pc/lab07$ nasm -f elf lab7-2.asm
user@daborovikov:~/work/arch-pc/lab07$ ld -m elf_i386 -o lab7-2 lab7-2.o
user@daborovikov:~/work/arch-pc/lab07$ ./lab7-2
106
user@daborovikov:~/work/arch-pc/lab07$
```

Рис. 2.7: Запуск исполняемого файла

Аналогично предыдущему примеру изменим символы на числа. Заменяем строки

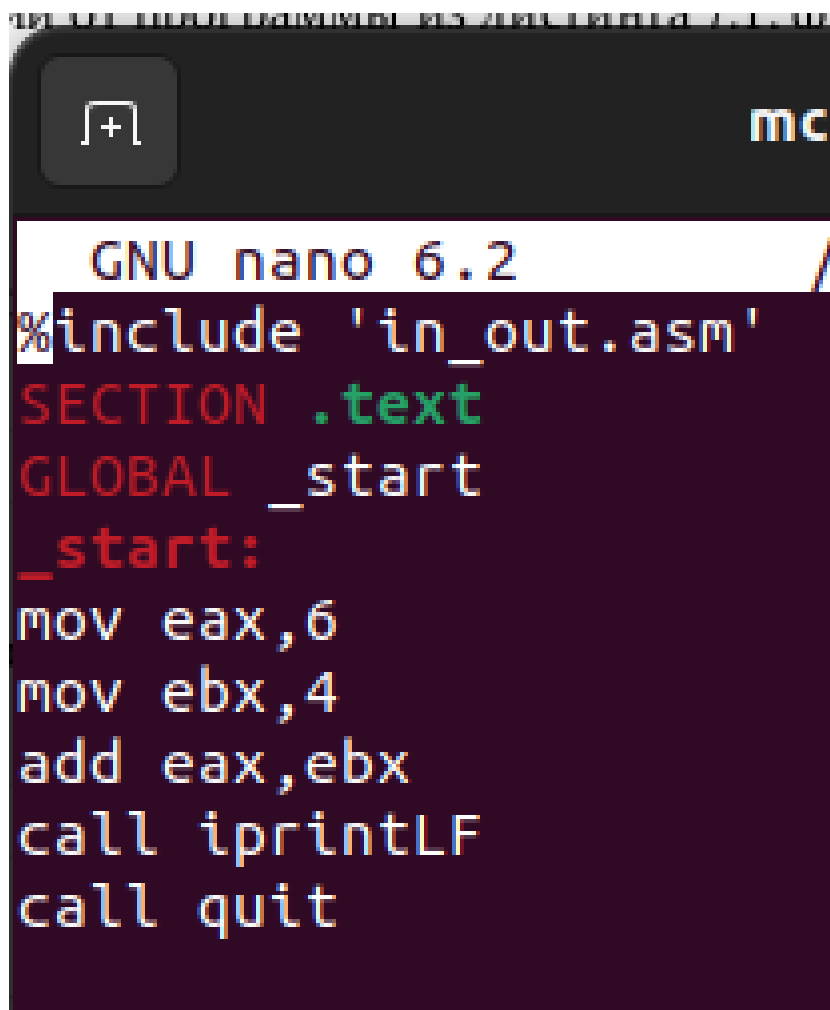
mov eax,'6'

mov ebx,'4'

на строки

moveax,6

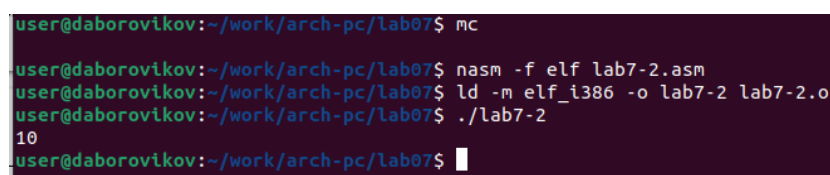
movebx,4(рис. 2.8)



```
GNU nano 6.2 /
%include 'in_out.asm'
SECTION .text
GLOBAL _start
_start:
mov eax,6
mov ebx,4
add eax,ebx
call iprintLF
call quit
```

Рис. 2.8: Замена символов на числа

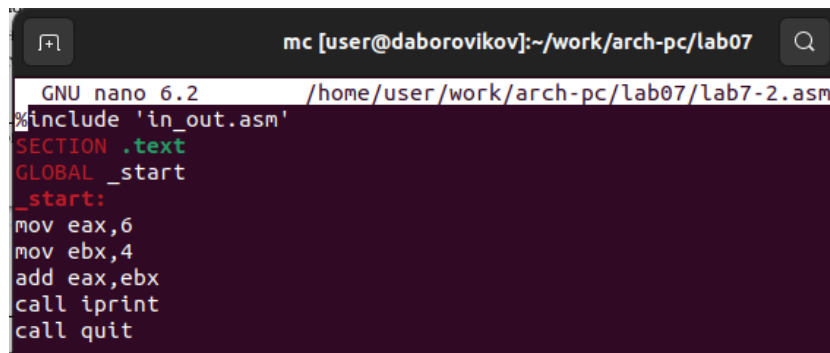
Создадим измененный исполняемый файл и запустим его(рис. 2.9)



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

Рис. 2.9: Запуск измененного исполняемого файла

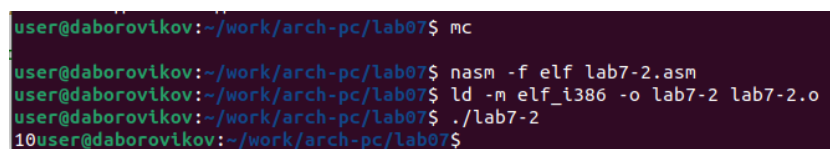
Заменяем функцию iprintLF на iprint(рис. 2.10)



```
mc [user@daborovikov]:~/work/arch-pc/lab07
GNU nano 6.2 /home/user/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 iprint
call quit
```

Рис. 2.10: Замена функции iprintLF на iprint

Создадим измененный исполняемый файл и запустим его(рис. 2.11)

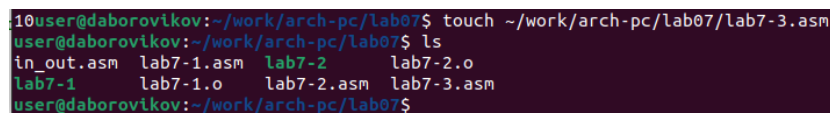


```
user@daborovikov:~/work/arch-pc/lab07$ mc
user@daborovikov:~/work/arch-pc/lab07$ nasm -f elf lab7-2.asm
user@daborovikov:~/work/arch-pc/lab07$ ld -m elf_i386 -o lab7-2 lab7-2.o
user@daborovikov:~/work/arch-pc/lab07$ ./lab7-2
10user@daborovikov:~/work/arch-pc/lab07$
```

Рис. 2.11: Запуск файла с iprint вместо iprintLF

Вывод функций iprintLF от iprint отличается отсутствием перевода на новую строку

Создадим файл lab7-3.asm в каталоге ~/work/arch-pc/lab07:(рис. 2.12)



```
10user@daborovikov:~/work/arch-pc/lab07$ touch ~/work/arch-pc/lab07/lab7-3.asm
user@daborovikov:~/work/arch-pc/lab07$ ls
in_out.asm  lab7-1.asm  lab7-2      lab7-2.o
lab7-1      lab7-1.o    lab7-2.asm  lab7-3.asm
user@daborovikov:~/work/arch-pc/lab07$
```

Рис. 2.12: Создание файла lab7-3.asm

Введем текст программы из листинга 7.3 в lab7-3.asm.(рис. 2.13)

```

mc [user@daborovikov]:~/work/arch-pc/lab07
GNU nano 6.2 /home/user/work/arch-pc/lab07/lab7-3.asm
%include 'in_out.asm' ; подключение внешнего файла
SECTION .data
div: DB 'Результат: ',0
rem: DB 'Остаток от деления: ',0
SECTION .text
GLOBAL _start
_start:
; ---- Вычисление выражения
mov eax,5 ; EAX=5
mov ebx,2 ; EBX=2
mul ebx ; EAX=EAX*EBX
add eax,3 ; EAX=EAX+3
xor edx,edx ; обнуляем EDX для корректной работы div
mov ebx,3 ; EBX=3
div ebx ; EAX=EAX/3, EDX=остаток от деления
mov edi,eax ; запись результата вычисления в 'edi'
; ---- Вывод результата на экран
mov eax,div ; вызов подпрограммы печати
call sprint ; сообщения 'Результат: '
mov eax,edi ; вызов подпрограммы печати значения
call iprintLF ; из 'edi' в виде символов
mov eax,rem ; вызов подпрограммы печати
call sprint ; сообщения 'Остаток от деления: '
mov eax,edx ; вызов подпрограммы печати значения
call iprintLF ; из 'edx' (остаток) в виде символов
call quit ; вызов подпрограммы завершения

```

Рис. 2.13: Текст программы lab7-3.asm

Создадим исполняемый файл и запустим его(рис. 2.14)

```

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

```

Рис. 2.14: Запуск исполняемого файла lab7-3.asm

Изменим текст программы для вычисления выражения $\boxtimes(\boxtimes) = (4 \boxtimes 6 + 2)/5$. (рис. 2.15)

```

user@daborovikov: ~/work/arch-pc/lab07
GNU nano 6.2 /home/user/work/arch-pc/lab07/lab7-3.asm *
%include 'in_out.asm' ; подключение внешнего файла
SECTION .data
div: DB 'Результат: ',0
rem: DB 'Остаток от деления: ',0
SECTION .text
GLOBAL _start
_start:
; ---- Вычисление выражения
mov eax,4 ; EAX=5
mov ebx,6 ; EBX=2
mul ebx ; EAX=EAX*EBX
add eax,2 ; EAX=EAX+3
xor edx,edx ; обнуляем EDX для корректной работы div
mov ebx,5 ; EBX=3
div ebx ; EAX=EAX/3, EDX=остаток от деления
mov edi,eax ; запись результата вычисления в 'edi'
; ----Вывод результата на экран
mov eax,div ; вызов подпрограммы печати
call sprint ; сообщения 'Результат: '
mov eax,edi ; вызов подпрограммы печати значения
call iprintLF ; из 'edi' в виде символов
mov eax,rem ; вызов подпрограммы печати
call sprint ; сообщения 'Остаток от деления: '
mov eax,edx ; вызов подпрограммы печати значения
call iprintLF ; из 'edx' (остаток) в виде символов
call quit ; вызов подпрограммы завершения

```

Рис. 2.15: Текст программы для вычисления выражения $\text{X}(\text{X}) = (4 \times 6 + 2)/5$.

Создадим исполняемый файл и запустим его(рис. 2.16)

```

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

```

Рис. 2.16: Запуск исполняемого файла для вычисления выражения $\text{X}(\text{X}) = (4 \times 6 + 2)/5$

Введем текст программы из листинга 7.4 в файлvariant.asm.(рис. 2.17)

```
user@daborovikov: ~/work/arch-pc/lab07
GNU nano 6.2 /home/user/work/arch-pc/lab07/variant.asm
#include 'in_out.asm'

SECTION .data
msg: DB 'Введите № студенческого билета: ',0
rem: DB 'Ваш вариант: ',0

SECTION .bss
x: RESB 80
SECTION .text
GLOBAL start
_start:

    mov eax, msg
    call sprintLF

    mov ecx, x
    mov edx, 80
    call sread

    mov eax, x
    call atoi

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

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

    call quit
```

Рис. 2.17: Листинг программы variant.asm

Создадим исполняемый файл и запустим его(рис. 2.18)

```
user@daborovikov:~/work/arch-pc/lab07$ nasm -f elf variant.asm
user@daborovikov:~/work/arch-pc/lab07$ ld -m elf_i386 -o variant variant.o
user@daborovikov:~/work/arch-pc/lab07$ ^C
user@daborovikov:~/work/arch-pc/lab07$ ./variant
Введите № студенческого билета:
113222006
Ваш вариант: 7
```

Рис. 2.18: Запуск исполняемого файла для нахождения номера варианта

Мой вариант номер 7

3 Ответы на вопросы

1. Какие строки листинга 7.4 отвечают за вывод на экран сообщения ‘Ваш вариант:’?

```
rem:DB ‘Ваш вариант:’,0  
mov eax,rem  
call sprint
```

2. Для чего используются следующие инструкции? `mov edx, 80` `mov ecx, x` `call sread`

Данные инструкции используются для ввода варианта(или другого значения) в переменную x

3. Для чего используется инструкция “`call atoi`”?

Преобразование кода ASCII в число

4. Какие строки листинга 7.4 отвечают за вычисления варианта?

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

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

В регистр edx

6. Для чего используется инструкция “inc edx”?

Прибавление единицы к регистру edx

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

```
mov eax,edx
```

```
call iprintLF
```

4 Самостоятельная работа

Создадим файл sam.asm и введем текст программы для вычисления выражения (вариант 7) - $5(x-1)^2$ (рис. 4.1)

```
1 %include 'in_out.asm'
2
3     SECTION .data
4
5     msg: DB 'Программа вычисления f(x)=5(x-1)^2 ',0
6     rem: DB 'Ответ: ',0
7
8     SECTION .bss
9     x: RESB 80
10
11    SECTION .text
12    GLOBAL _start
13    _start:
14
15    mov eax, msg
16    call sprintf
17
18    mov ecx, x
19    mov edx, 80
20    call sread
21
22    mov eax, x
23    call atoi
24
25    add eax, -1
26
27    mov ebx, eax
28    mul ebx
29
30    mov ebx, 5
31    mul ebx
32
33    mov edi, eax
34
35    mov eax, rem
36    call sprintf
37    mov eax, edi
38    call iprintLF
39
40    call quit
```

Рис. 4.1: Листинг программы sam.asm

Создадим исполняемый файл и проверим работу на значениях $x_1=3$ и $x_2=5$.

Проверим ответы программы ручных вычислением. Всё работает верно (рис. 4.2)

```
user@daborovikov:~/work/arch-pc/lab07$ nasm -f elf sam.asm
user@daborovikov:~/work/arch-pc/lab07$ ld -m elf_i386 -o sam sam.o
user@daborovikov:~/work/arch-pc/lab07$ ./sam
Программа вычисления  $f(x)=5(x-1)^2$ 
3
Ответ: 20
user@daborovikov:~/work/arch-pc/lab07$ ./sam
Программа вычисления  $f(x)=5(x-1)^2$ 
5
Ответ: 80
user@daborovikov:~/work/arch-pc/lab07$
```

Рис. 4.2: Запуск исполняемого файла для решения выражения $5(x-1)^2$

5 Выводы

В ходе лабораторной работы мы освоили арифметические инструкции языка ассемлера NASM.

https://github.com/daBorovikov/study_2022-2023_arh-pc-