

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра прикладной информатики и теории вероятностей

ОТЧЕТ

ПО ЛАБОРАТОРНОЙ РАБОТЕ № 6

дисциплина: *Архитектура компьютера*

Студент: Кошина Анастасия

Группа: НБИбд-04-20

МОСКВА

2022 г.

Цель работы:

Освоение арифметических инструкций языка ассемблера NASM.

Порядок выполнения работы:

1. Создайте каталог для программ лабораторной работы № 7, перейдите в него и создайте файл lab7-1.asm.
2. Рассмотрим примеры программ вывода символьных и численных значений. Программы будут выводить значения, записанные в регистр
3. Далее изменим текст программы и вместо символов, запишем в регистры числа. Исправьте текст программы следующим образом:

замените строки mov eax,'6' mov ebx,'4'

на строки mov eax,6 mov ebx,4

Создайте исполняемый файл и запустите его.

Как и в предыдущем случае при исполнении программы мы не получим число 10. В данном случае выводится символ с кодом 10. Пользуясь таблицей ASCII определите какому символу соответствует код 10.

Отображается ли этот символ при выводе на экран?

4. Как отмечалось выше, для работы с числами в файле in_out.asm реализованы подпрограммы для преобразования ASCII символов в числа и обратно. Преобразуем текст программы из Листинга с использованием этих функций.

Создайте файл lab7-2.asm в каталоге ~/work/arch-pc/lab07 и введите в него текст программы из листинга.

Создайте исполняемый файл и запустите его.

5. Аналогично предыдущему примеру изменим символы на числа.

Замените строки mov eax,'6' mov ebx,'4'

на строки mov eax,6 mov ebx,4

Создайте исполняемый файл и запустите его. Какой результат будет

получен при исполнении программы? Замените функцию `iprintLF` на `iprint`. Создайте исполняемый файл и запустите его. Чем отличается вывод функций `iprintLF` и `iprint`?

6. В качестве примера выполнения арифметических операций в NASM приведем программу вычисления арифметического выражения:

$$f(x) = (5 * 2 + 3)/3.$$

Создайте файл `lab7-3.asm` в каталоге `~/work/arch-pc/lab07`.

Внимательно изучите текст программы из листинга и введите в `lab7-`

Создайте исполняемый файл и запустите его.

Измените текст программы для вычисления выражения

$$f(x) = (4 * 6 + 2)/5.$$

Создайте исполняемый файл и проверьте его работу.

7. В качестве другого примера рассмотрим программу вычисления варианта задания по номеру студенческого билета, работающую по следующему алгоритму:

- вывести запрос на введение № студенческого билета.
- вычислить номер варианта по формуле: $(Sn \bmod 20) + 1$, где Sn – номер студенческого билета (в данном случае $a \bmod b$ – это остаток от деления a на b).
- вывести на экран номер варианта.

Создайте файл `variant.asm` в каталоге `~/work/arch-pc/lab07`:

Внимательно изучите текст программы из листинга и введите в файл

Создайте исполняемый файл и запустите его. Проверьте результат работы программы вычислив номер варианта аналитически.

Включите в отчет по выполнению лабораторной работы ответы на следующие вопросы:

1. Какие строки листинга 7.4 отвечают за вывод на экран сообщения ‘Ваш вариант:’?
2. Для чего используются следующие инструкции?
`mov ecx, x mov edx, 80 call sread`
3. Для чего используется инструкция “call atoi”?
4. Какие строки листинга 7.4 отвечают за вычисления варианта?
5. В какой регистр записывается остаток от деления при выполнении инструкции “div ebx”?
6. Для чего используется инструкция “inc edx”?
7. Какие строки листинга 7.4 отвечают за вывод на экран результата вычислений?

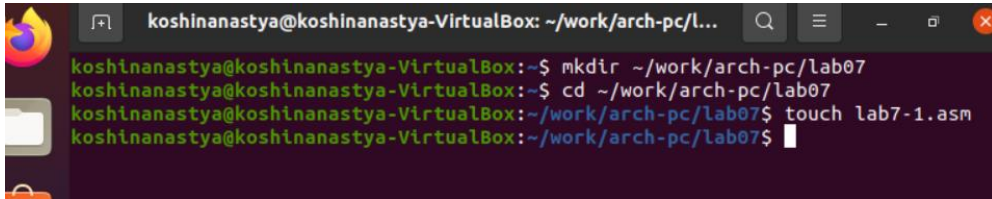
Задание для самостоятельной работы:

1. Написать программу вычисления выражения $y = f(x)$. Программа должна выводить выражение для вычисления, выводить запрос на ввод значения x , вычислять заданное выражение в зависимости от введенного x , выводить результат вычислений. Вид функции $f(x)$ выбрать из таблицы вариантов заданий в соответствии с номером, полученным при выполнении лабораторной работы. Создайте исполняемый файл и проверьте его работу для значений x_1 и x_2 .

При выполнении задания преобразовывать (упрощать) выражения для $f(x)$ нельзя. При выполнении деления в качестве результата можно использовать только целую часть от деления и не учитывать остаток (т.е. $5 : 2 = 2$).

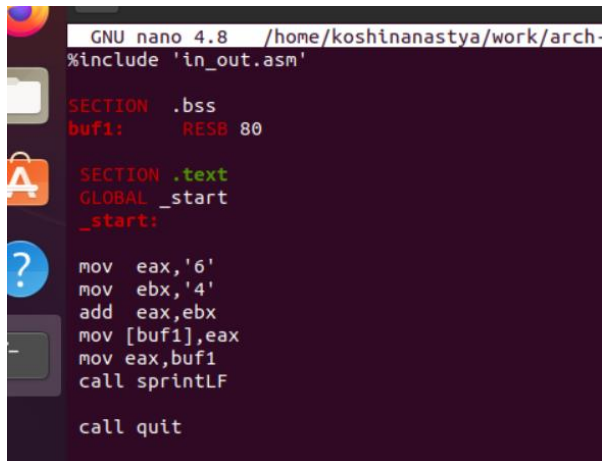
Ход работы:

1. Я создала каталог для программ лабораторной работы № 7, перешла в него и создала файл lab7-1.asm.



```
koshinanastya@koshinanastya-VirtualBox: ~/work/arch-pc/l...
koshinanastya@koshinanastya-VirtualBox:~$ mkdir ~/work/arch-pc/lab07
koshinanastya@koshinanastya-VirtualBox:~$ cd ~/work/arch-pc/lab07
koshinanastya@koshinanastya-VirtualBox:~/work/arch-pc/lab07$ touch lab7-1.asm
koshinanastya@koshinanastya-VirtualBox:~/work/arch-pc/lab07$
```

2. Затем ввела в файл lab7-1.asm текст программы из листинга 7.1:



```
GNU nano 4.8 /home/koshinanastya/work/arch-
%include 'in_out.asm'

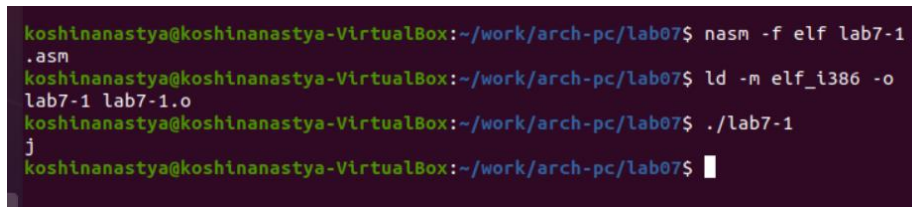
SECTION .bss
buf1:  RESB 80

SECTION .text
GLOBAL _start
_start:

mov  eax,'6'
mov  ebx,'4'
add  eax,ebx
mov  [buf1],eax
mov  eax,buf1
call sprintLF

call quit
```

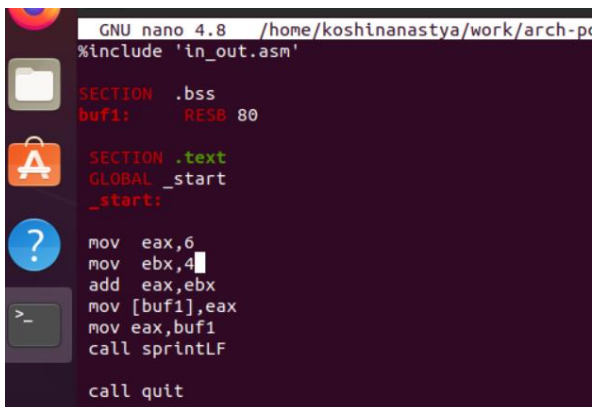
Создала исполняемый файл и запустила его:



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

3. Далее изменила текст программы:

заменяла строки `mov eax,'6'` `mov ebx,'4'`
на строки `mov eax,6` `mov ebx,4`



```
GNU nano 4.8 /home/koshinanastya/work/arch-pc
%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
```

Создала исполняемый файл и запустила его:

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

koshinanastya@koshinanastya-VirtualBox:~/work/arch-pc/lab07$
```

4. Создала файл lab7-2.asm в каталоге ~/work/arch-pc/lab07 и ввела в него текст программы из листинга 7.2.

```
koshinanastya@koshinanastya-VirtualBox:~/work/arch-pc/lab07$ touch lab7-2.asm
koshinanastya@koshinanastya-VirtualBox:~/work/arch-pc/lab07$
```

```
GNU nano 4.8 /home/koshinanastya/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
```

Создала исполняемый файл и запустила его:

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

5. Аналогично предыдущему примеру изменила символы на числа:

заменяла строки mov eax,'6' mov ebx,'4'

на строки mov eax,6 mov ebx,4

```
GNU nano 4.8 /home/koshinanastya/work/arch-pc/lab07/lab7-2.asm  Изменён
#include 'in_out.asm'

SECTION .text
GLOBAL _start
_start:

mov  eax,6
mov  ebx,4
add  eax,ebx
call iprintLF

call quit
```

Создала исполняемый файл и запустила его:

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

6. В качестве примера выполнения арифметических операций в NASM привела программу вычисления арифметического выражения:

$$f(x) = (5 * 2 + 3)/3.$$

Создала файл lab7-3.asm в каталоге ~/work/arch-pc/lab07:

```
10
koshinanastya@koshinanastya-VirtualBox:~/work/arch-pc/lab07$ touch lab7-3.asm
koshinanastya@koshinanastya-VirtualBox:~/work/arch-pc/lab07$
```

Внимательно изучила текст программы из листинга 7.3 и ввела в lab7-3.asm.

```
GNU nano 4.8 /home/koshinanastya/work/arch-pc/lab07/lab7-3.asm
%include 'in_out.asm'

SECTION .data
div: DB 'Результат: ',0
rem: DB ' Остаток от деления: ',0

SECTION .text
GLOBAL _start
_start:

mov eax,5
mov ebx,2
mul ebx
add eax,3
xor edx,edx
mov ebx,3
div ebx

mov edi,eax

mov eax,div
call sprint
mov eax,edi
call iptintLF

^O Помощь ^O Записать ^W Поиск ^K Вырезать ^J Вывести
^X Выход ^R ЧитФайл ^L Замена ^U Paste Text ^T Словари
```

Создала исполняемый файл и запустила его:

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

Изменила текст программы для вычисления выражения:

$$f(x) = (4 * 6 + 2)/5:$$

```
GNU nano 4.8 /home/koshinanastya/work/arch-pc/lab07/lab7-3.asm Изменен
%include 'in_out.asm'

SECTION .data
div: DB 'Результат: ',0
rem: DB ' Остаток от деления: ',0

SECTION .text
GLOBAL _start
_start:

mov eax,6
mov ebx,4
mul ebx
add eax,2
xor edx,edx
mov ebx,5
div ebx

mov edi,eax

mov eax,div
call sprint
mov eax,edi
call iptintLF

^O Помощь ^O Записать ^W Поиск ^K Вырезать ^J Вывести
^X Выход ^R ЧитФайл ^L Замена ^U Paste Text ^T Словари
```

Создала исполняемый файл и проверила его работу:

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

7. В качестве другого примера рассмотрела программу вычисления варианта задания по номеру студенческого билета, работающую по следующему алгоритму:

- вывести запрос на введение № студенческого билета.
- вычислить номер варианта по формуле: $(Sn \bmod 20) + 1$, где Sn – номер студенческого билета (в данном случае $a \bmod b$ – это остаток от деления a на b).
- вывести на экран номер варианта.

Создала файл variant.asm в каталоге ~/work/arch-pc/lab07:

```
Остаток от деления: 1
koshinanastya@koshinanastya-VirtualBox:~/work/arch-pc/lab07$ touch variant.asm
koshinanastya@koshinanastya-VirtualBox:~/work/arch-pc/lab07$
```

Внимательно изучила текст программы из листинга 7.4 и ввела в файл

```
GNU nano 4.8 /home/koshinanastya/work/arch-pc/lab07/variant.asm Изменён
#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
```

Создала исполняемый файл и запустила его:

```
koshinanastya@koshinanastya-VirtualBox:~/work/arch-pc/lab07$ nasm -f elf variant.asm
koshinanastya@koshinanastya-VirtualBox:~/work/arch-pc/lab07$ ld -m elf_i386 -o variant variant.o
koshinanastya@koshinanastya-VirtualBox:~/work/arch-pc/lab07$ ./variant
Введите № студенческого билета:
1032225861
Ваш вариант: 2
koshinanastya@koshinanastya-VirtualBox:~/work/arch-pc/lab07$
```


Вопросы и ответы:

1. Какие строки листинга 7.4 отвечают за вывод на экран сообщения

‘Ваш вариант: ’?

r

2. Для чего используются следующие инструкции?

mov ecx, x mov edx, 80 call sread

DB 'Ваш вариант: ',0

n

0

3. Для чего используется инструкция “call atoi”?

Преобразовывает строку в целое число.

4. Какие строки листинга 7.4 отвечают за вычисления варианта?

inteadеактивнов подпрограммы ввода сообщения

call iprintбщения.мы.

mov eax,edx

call iprintLF

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

E

6. Для чего используется инструкция “inc edx”?

X

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

edx увеличивает значение регистра edx на 1

a

1

1

iprintLF

Задание для самостоятельной работы:

Написала программу вычисления выражения $y = f(x)$. Вид функции $f(x)$ выбрала из таблицы вариантов заданий в соответствии с номером, полученным при выполнении лабораторной работы.

В моем случае это вариант №2

Функция: $(12x + 3)5$

X

X

Для выполнения функции с X1, создала файл lab7_4.asm:

```
Ваш вариант: 2
koshinanastya@koshinanastya-VirtualBox:~/work/arch-pc/lab07$ touch lab7_4.asm
koshinanastya@koshinanastya-VirtualBox:~/work/arch-pc/lab07$
```

и ввела в него:

```
GNU nano 4.8 /home/koshinanastya/work/arch-pc/lab07/lab7-4.asm Изменен
#include "in_out.asm"

SECTION .data
div: DB 'Результат: ',0
rem: DB ' Остаток от деления: ',0

SECTION .text
GLOBAL _start
_start:

mov eax,12
mov ebx,1
mul ebx
add eax,3
xor edx,edx
mov ebx,5
mul ebx

mov edi,eax

mov eax,div
call sprint
mov eax,edi
call iprintf

^G Помощь ^O Записать ^N Поиск ^K Вырезать ^J Выводить
^X Выход ^R ЧитФайл ^\ Замена ^U Paste Text ^T Словарь
```

Для выполнения функции с X2, создала файл lab7_5.asm:

```
koshinanastya@koshinanastya-VirtualBox:~/work/arch-pc/lab07$ touch lab7_5.asm
koshinanastya@koshinanastya-VirtualBox:~/work/arch-pc/lab07$
```

и ввела в него:

```
GNU nano 4.8 /home/koshinanastya/work/arch-pc/lab07/lab7-5.asm Изменен
#include "in_out.asm"

SECTION .data
div: DB 'Результат: ',0
rem: DB ' Остаток от деления: ',0

SECTION .text
GLOBAL _start
_start:

mov eax,12
mov ebx,6
mul ebx
add eax,3
xor edx,edx
mov ebx,5
mul ebx

mov edi,eax

mov eax,div
call sprint
mov eax,edi
call iprintf

Имя файла для записи: /home/koshinanastya/work/arch-pc/lab07/lab7-5.asm
```

Проверила работу файлов lab7-4.asm и lab7-5.asm:

```
koshinanastya@koshinanastya-VirtualBox:~/work/arch-pc/lab07$ nasm -f elf lab7-4
.asm
koshinanastya@koshinanastya-VirtualBox:~/work/arch-pc/lab07$ ld -m elf_i386 -o
lab7-4 lab7-4.o
koshinanastya@koshinanastya-VirtualBox:~/work/arch-pc/lab07$ ./lab7-4
Результат: 75
Остаток от деления: 0
koshinanastya@koshinanastya-VirtualBox:~/work/arch-pc/lab07$ nasm -f elf lab7-5
.asm
koshinanastya@koshinanastya-VirtualBox:~/work/arch-pc/lab07$ ld -m elf_i386 -o
lab7-5 lab7-5.o
koshinanastya@koshinanastya-VirtualBox:~/work/arch-pc/lab07$ ./lab7-5
Результат: 375
Остаток от деления: 0
koshinanastya@koshinanastya-VirtualBox:~/work/arch-pc/lab07$
```

Вывод:

С помощью этой лабораторной работы, я освоила арифметические инструкции языка ассемблера NASM.