

Лабораторная работа №7

Руслан Шухратович Исмаилов

Содержание

1	Цель работы	4
2	Задание	5
3	Выполнение лабораторной работы	6
4	Ответы на вопросы 1-7	8
5	Задание для самостоятельной работы	9
6	Выводы	14

Список иллюстраций

5.1	lab7-1	10
5.2	lab7-1	10
5.3	lab7-2	10
5.4	lab7-2	10
5.5	lab7-2	10
5.6	lab7-3	11
5.7	отредактированный код lab7-3.asm	11
5.8	ответ lab7-3	11
5.9	вариант	12
5.10	ответ	12
5.11	код fx	13

1 Цель работы

Освоение арифметических инструкций языка ассемблера NASM.

2 Задание

Написать программу для вычисления функции из одного из вариантов, ознакомиться с Арифметическими операциями на языке NASM

3 Выполнение лабораторной работы

Шаг 1

Создадим каталог лабораторной работы 7, и в нём создадим файл lab7-1.asm (не забывая перенести туда файл in_out.asm), перенесём в него текст программы 7.1. Создадим исполняемый файл и увидим символ 'j', так как сложение кодов символов 4 и 6 равняется коду символа j (в ASCII)

(рис. 5.1)

Шаг 2

Изменим текст файла, заменим

```
mov eax, '6'
```

```
mov ebx, '4'
```

на

```
mov eax, 6
```

```
mov ebx, 4
```

Вместо 106, получим 10 (6+4) символ ASCII, который соответствует символу 'LF,/n', (не отображается на экране.)

(рис. 5.2)

Шаг 3

Создадим файл lab7-2.asm, где мы используем код из файла in_out.asm для преобразования ASCII символов в числа. Копируем код соответственного листинга 7.2, запустим исполняемый файл, :

(рис. 5.3)

Получим число 106, сумму кодов символов '4' и '6'
снова заменим

```
mov eax, '6'  
mov ebx, '4'
```

на

```
mov eax, 6  
mov ebx, 4
```

и, запустив файл, получим число 10 (6+4)

(рис. 5.4)

Также заменим в исходном файле функцию `iprintLF` на `iprint`
после вывода результата не переводится строка терминала

(рис. 5.5)

Шаг 4

В файл `lab7-3.asm` введём код из листинга 7.3

найдем выражение $((5 \times 2) + 3) / 3$

Создадим файл и получим ответ:

(рис. 5.6)

Изменим код файла для вычисления выражения $((4 \times 6) + 2) / 5$

(рис. 5.7)

Ответ:

(рис. 5.8)

Шаг 5

вводим в `variant.asm` код программы 7.4, проверяем его работу:

(рис. 5.9)

Убедимся в правильности результата, разделим 1032225753 на 20, получим 51611287 и 13 в остатке, прибавим к остатку 1, это равняется номеру варианта

4 Ответы на вопросы 1-7

1.

```
mov eax, msg
```

```
call sprintf
```

Отвечает за вывод сообщения 'ваш вариант: ...'

2. Данные функции используются для ввода значения переменной X с клавиатуры и сохранения введенных данных

3. Данная инструкция используется для преобразования кода символа ASCII в число

4.

```
mov ebx, 20
```

```
div ebx
```

```
inc edx
```

5. В регистр edx

6. Для увеличения значения edx на +1.

7. Результат вычислений:

```
mov eax, edx
```

```
call iprintLF
```


5 Задание для самостоятельной работы

Шаг 1

Создадим файл fx.asm, где решим $(x/2+8)^3$ в зависимости от различных X.

Рассмотрим принцип работы нашего кода

```
mov eax, msg
call sprintf
mov ecx, x
mov edx, 80
call sread
mov eax, x
call atoi
xor edx, edx
```

отвечает за запись переменной X

```
mov ebx, 2 ; ebx=2
idiv ebx ; eax / 2
```

Деление X на ebx (2)

```
add eax, 8 ; eax + 8
```

Сложение X + 8

```
mov ebx, 3 ; ebx=3
mul ebx ; eax * ebx
```

умножение X на ebx (3)

Запишем результат в edi, выведем на экран с помощью iprintLF

Код: (рис. 5.11)

Проверим: (рис. 5.10)

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

Рис. 5.1: lab7-1

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

Рис. 5.2: lab7-1

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

Рис. 5.3: lab7-2

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

Рис. 5.4: lab7-2

```
[rsismailov@fedora lab07]$ ./lab7-2
10[rsismailov@fedora lab07]$
```

Рис. 5.5: lab7-2

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

Рис. 5.6: lab7-3

```
%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
```

Рис. 5.7: отредактированный код lab7-3.asm

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

Рис. 5.8: ответ lab7-3

```
[rsismailov@fedora lab07]$ touch variant.asm
[rsismailov@fedora lab07]$ nasm -f elf variant.asm
[rsismailov@fedora lab07]$ ld -m elf_i386 -o variant variant.o
[rsismailov@fedora lab07]$ ./variant
Введите No студенческого билета:
1032225753
Ваш вариант: 14
```

Рис. 5.9: вариант

```
[rsismailov@fedora lab07]$ ./fx
Введите X:
1
Результат: 24
[rsismailov@fedora lab07]$ ./fx
Введите X:
4
Результат: 30
```

Рис. 5.10: ответ

```

%include 'in_out.asm' ; подключение внешнего файла

SECTION .data
msg: DB 'Введите X: ',0
div: DB 'Результат: ',0
SECTION .bss
x: RESB 80
SECTION .text
GLOBAL _start
_start:

;Вычисление выражения (x/2+8)*3

mov eax, msg
call sprintf
mov ecx, x
mov edx, 80
call sread
mov eax,x
call atoi
xor edx,edx

mov ebx,2 ; ebx=2
idiv ebx ; eax / 2

add eax,8 ; eax + 8

mov ebx,3 ; ebx=3
mul ebx ; eax * ebx

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

call quit ; вызов подпрограммы завершения

```

Рис. 5.11: код fx

6 Выводы

Я написал программу для вычисления функции и ознакомился с принципом вычисления на языке ассемблера NASM