

Отчёт по лабораторной работе №6

Кучерова Виктория Васильевна

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
3	Задание для самостоятельной работы	16
4	Выводы	19
	Список литературы	20

Список иллюстраций

2.1	Создание файла	6
2.2	Програма1	7
2.3	Запуск1	7
2.4	Програма2	8
2.5	Запуск2	8
2.6	Создание файла	9
2.7	Програма3	9
2.8	Запуск3	9
2.9	Програма4	10
2.10	Запуск4	10
2.11	Програма5	11
2.12	Запуск5	11
2.13	Создание файла	11
2.14	Програма6	12
2.15	Запуск6	12
2.16	Програма7	13
2.17	Запуск7	13
2.18	Создание файла	13
2.19	Програма8	14
2.20	Запуск8	14
3.1	Програма9	17
3.2	Запуск9	18

Список таблиц

1 Цель работы

Освоить арифметические инструкции языка ассемблера NASM.

2 Выполнение лабораторной работы

Создадим каталог для программ лабораторной работы № 6, перейдем в него и создадим файл lab6-1.asm(рис. 2.1).

```
vvkucheroval@vbox:~$ mkdir ~/work/arch-pc/lab06  
vvkucheroval@vbox:~$ cd ~/work/arch-pc/lab06  
vvkucheroval@vbox:~/work/arch-pc/lab06$ touch lab6-1.asm
```

Рис. 2.1: Создание файла

Рассматриваем примеры программ вывода символьных и численных значений. Программы будут выводить значения записанные в регистрах eax. Создаем исполняемый файл и запускаем его(рис. 2.2), (рис. 2.3).

```

%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. 2.2: Програма1

```

vvkucheroval@vbox:~/work/arch-pc/lab06$ nasm -f elf lab6-1.asm
vvkucheroval@vbox:~/work/arch-pc/lab06$ ld -m elf_i386 -o lab6-1 lab6-1.o
vvkucheroval@vbox:~/work/arch-pc/lab06$ ./lab6-1
j

```

Рис. 2.3: Запуск1

Далее изменяем текст программы и вместо символов, запишем в регистры числа(рис. 2.4), (рис. 2.5).

```

%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. 2.4: Програма2

```

vvkucheroval@vbox:~/work/arch-pc/lab06$ nasm -f elf lab6-1.asm
vvkucheroval@vbox:~/work/arch-pc/lab06$ ld -m elf_i386 -o lab6-1 lab6-1.o
vvkucheroval@vbox:~/work/arch-pc/lab06$ ./lab6-1

```

Рис. 2.5: Запуск2

Создаем файл lab6-2.asm в каталоге ~/work/arch-pc/lab06 и вводим в него текст программы(рис. 2.6), (рис. 2.7), (рис. 2.8).


```
vvkucheroval@vbox:~/work/arch-pc/lab06$ touch lab6-2.asm
```

Рис. 2.6: Создание файла

```
%include 'in_out.asm'

SECTION .text
GLOBAL _start
_start:

    mov eax, '6'
    mov ebx, '4'
    add eax, ebx
    call iprintLF
    |
    call quit
```

Рис. 2.7: Програма3

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

Рис. 2.8: Запуск3

Аналогично предыдущему примеру изменяем символы на числа(рис. 2.9), (рис. 2.10).

```

%include 'in_out.asm'

SECTION .text
GLOBAL _start
_start:

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

call quit

```

Рис. 2.9: Програма4

```

vvkucheroва@vbox:~/work/arch-pc/lab06$ nasm -f elf lab6-2.asm
vvkucheroва@vbox:~/work/arch-pc/lab06$ ld -m elf_i386 -o lab6-2 lab6-2.o
vvkucheroва@vbox:~/work/arch-pc/lab06$ ./lab6-2
10

```

Рис. 2.10: Запуск4

Заменяем функцию iprintLF на iprint(рис. 2.11), (рис. 2.12).

```

%include 'in_out.asm'

SECTION .text
GLOBAL _start
_start:

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

call quit

```

Рис. 2.11: Програма5

```

vvkucheroval@vbox:~/work/arch-pc/lab06$ nasm -f elf lab6-2.asm
vvkucheroval@vbox:~/work/arch-pc/lab06$ ld -m elf_i386 -o lab6-2 lab6-2.o
vvkucheroval@vbox:~/work/arch-pc/lab06$ ./lab6-2
10vvkucheroval@vbox:~/work/arch-pc/lab06$

```

Рис. 2.12: Запуск5

В качестве примера выполнения арифметических операций в NASM приведем программу вычисления арифметического выражения $f(x) = (5 * 2 + 3)/3$ (рис. 2.13), (рис. 2.14), (рис. 2.15).

```

10vvkucheroval@vbox:~/work/arch-pc/lab06$ touch lab6-3.asm

```

Рис. 2.13: Создание файла

```

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

    call quit

```

Рис. 2.14: Програма6

```

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

```

Рис. 2.15: Запуск6

Изменяем текст программы для вычисления выражения $f(x) = (4 * 6 + 2)/5$. Создаем исполняемый файл и проверяем его работу(рис. 2.16), (рис. 2.17).

```

%include 'in_out.asm'

SECTION .data

div: DB 'Результат: ',0
rem: DB 'Остаток от деления: ',0

SECTION .text
GLOBAL _start
_start:

mov eax,4
mov ebx,6
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 iprintLF
mov eax,rem
call sprint
mov eax,edx
call iprintLF

call quit

```

Рис. 2.16: Програма7

```

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

```

Рис. 2.17: Запуск7

В качестве другого примера рассмотрим программу вычисления варианта задания по номеру студенческого билета(рис. 2.18), (рис. 2.19), (рис. 2.20).

```

vvkucheroва@vbox:~/work/arch-pc/lab06$ touch variant.asm

```

Рис. 2.18: Создание файла

```

%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

```

Рис. 2.19: Програма8

```

vvkucheroва@vbox:~/work/arch-pc/lab06$ nasm -f elf variant.asm
vvkucheroва@vbox:~/work/arch-pc/lab06$ ld -m elf_i386 -o variant variant.o
vvkucheroва@vbox:~/work/arch-pc/lab06$ ./variant
Введите № студенческого билета:
1132246778
Ваш вариант: 19

```

Рис. 2.20: Запуск8

Ответы на вопросы:

- 1 За вывод сообщения “Ваш вариант” отвечают строки кода: `mov eax,rem` `call sprint`
- 2 Инструкция `mov ecx, x` используется, чтобы положить адрес вводимой строки `x` в регистр `ecx` `mov edx, 80` - запись в регистр `edx` длины вводимой строки `call sread`

- вызов подпрограммы из внешнего файла, обеспечивающей ввод сообщения с клавиатуры

3 `call atoi` используется для вызова подпрограммы из внешнего файла, которая преобразует `ascii`-код символа в целое число и записывает результат в регистр `eax`

4 За вычисления варианта отвечают строки: `xor edx,edx` ; обнуление `edx` для корректной работы `divmov ebx,20` ; `ebx = 20 div ebx` ; `eax = eax/20`, `edx` - остаток от деления `inc edx` ; `edx = edx + 1`

5 При выполнении инструкции `div ebx` остаток от деления записывается в регистр `edx`

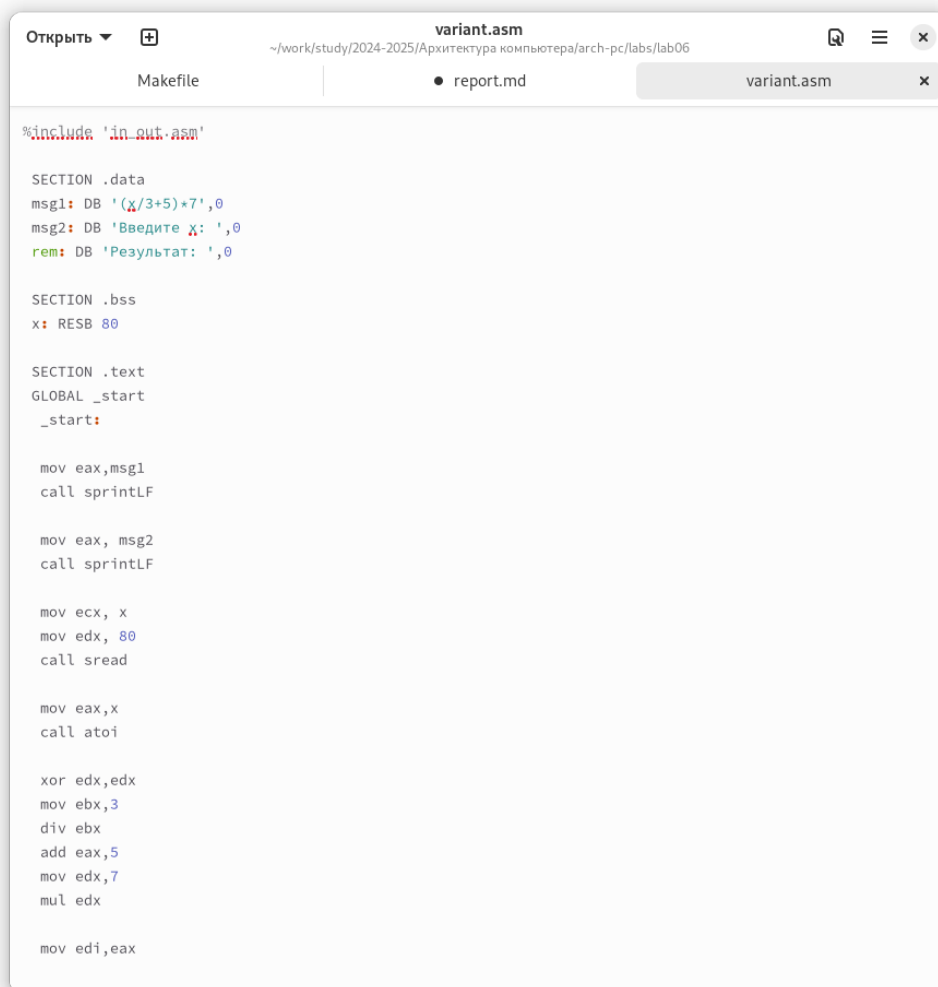
6 Инструкция `inc edx` увеличивает значение регистра `edx` на 1

7 За вывод на экран результатов вычислений отвечают строки:

```
mov eax,edx call iprintLF
```

3 Задание для самостоятельной работы

Напишем программу вычисления выражения $y = f(x)$ (рис. 3.2), (рис. 3.1).



The image shows a code editor window with the title bar 'variant.asm'. The address bar shows the path '~\work\study\2024-2025\Архитектура компьютера\arch-pc\labs\lab06'. The editor contains the following assembly code:

```
%include 'in_out.asm'

SECTION .data
msg1: DB '(\3+5)*7',0
msg2: DB 'Введите x: ',0
rem: DB 'Результат: ',0

SECTION .bss
x: RESB 80

SECTION .text
GLOBAL _start
_start:

mov eax,msg1
call sprintf

mov eax, msg2
call sprintf

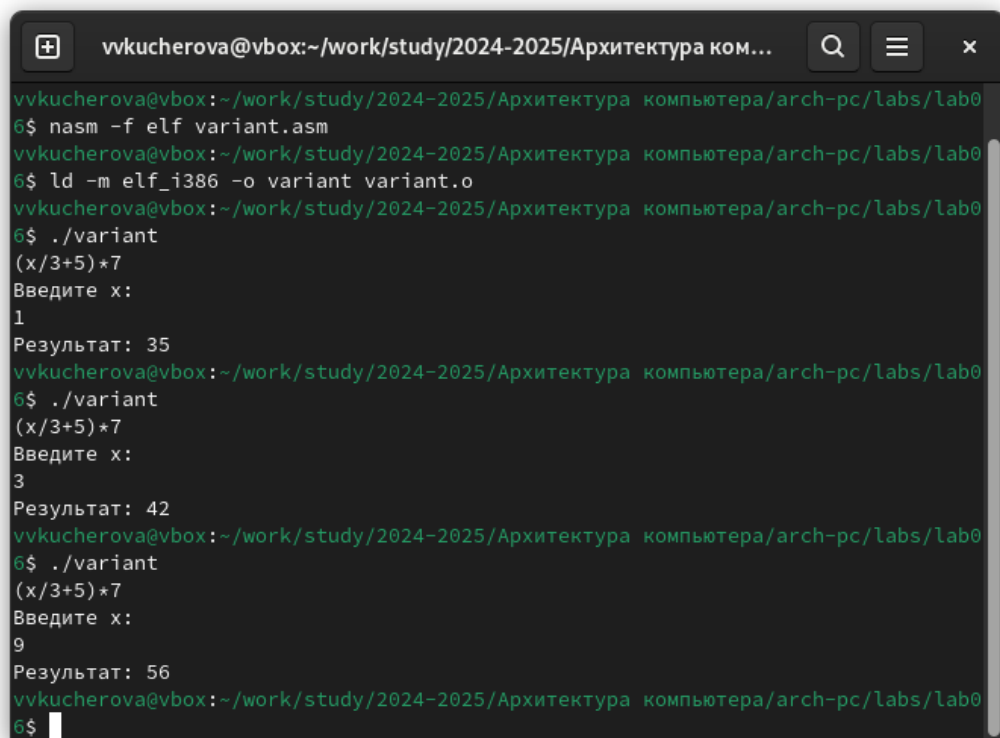
mov ecx, x
mov edx, 80
call sread

mov eax,x
call atoi

xor edx,edx
mov ebx,3
div ebx
add eax,5
mov edx,7
mul edx

mov edi,eax
```

Рис. 3.1: Програма9



```
vvkucheroва@vbox:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab0
6$ nasm -f elf variant.asm
vvkucheroва@vbox:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab0
6$ ld -m elf_i386 -o variant variant.o
vvkucheroва@vbox:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab0
6$ ./variant
(x/3+5)*7
Введите x:
1
Результат: 35
vvkucheroва@vbox:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab0
6$ ./variant
(x/3+5)*7
Введите x:
3
Результат: 42
vvkucheroва@vbox:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab0
6$ ./variant
(x/3+5)*7
Введите x:
9
Результат: 56
vvkucheroва@vbox:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab0
6$
```

Рис. 3.2: Запуск9

4 Выводы

Я освоила арифметические инструкции языка ассемблера NASM.

Список литературы