

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра прикладной информатики и теории вероятностей

ОТЧЕТ

ПО ЛАБОРАТОРНОЙ РАБОТЕ № 6

дисциплина: Архитектура вычислительных систем

Студент: Панченко Денис Дмитриевич

Группа: НБИбд-03-22

МОСКВА

2022 г.

Цель работы:

Освоить арифметические инструкции языка ассемблера NASM.

Ход работы:

1) Создаем каталог для программ лабораторной работы № 6, переходим в него и создаем файл lab6-1.asm. (Рис. 1)

```
ddpanchenko@nbibd-03-22:~$ mkdir ~/work/arch-pc/lab06
ddpanchenko@nbibd-03-22:~$ cd ~/work/arch-pc/lab06
ddpanchenko@nbibd-03-22:~/work/arch-pc/lab06$ touch lab6-1.asm
```

Рис. 1. Создание каталога с файлом

2) Рассмотрим примеры программ вывода символьных и численных значений.

Программы будут выводить значения записанные в регистр eax. (Рис. 2)

Создаем исполняемый файл и запускаем его. (Рис. 3)

```
GNU nano 4.8
%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. Текст программы

```
ddpanchenko@nbibd-03-22:~/work/arch-pc/lab06$ nasm -f elf lab6-1.asm
ddpanchenko@nbibd-03-22:~/work/arch-pc/lab06$ ld -m elf_i386 -o lab6-1 lab6-1.o
ddpanchenko@nbibd-03-22:~/work/arch-pc/lab06$ ./lab6-1
j
```

Рис. 3. Запуск файла

3) Далее изменяем текст программы и вместо символов, записываем в регистры числа. Исправляем текст программы. (Рис. 4)

Создаем исполняемый файл и запускаем его. (Рис. 5)

```
GLOBAL _start
_start:
mov eax,6
mov ebx,4
add eax,ebx
mov [buf1],eax
mov eax,buf1
```

Рис. 4. Текст программы

```
ddpanchenko@nb1bd-03-22:~/work/arch-pc/lab06$ nasm -f elf lab6-1.asm
ddpanchenko@nb1bd-03-22:~/work/arch-pc/lab06$ ld -m elf_i386 -o lab6-1 lab6-1.o
ddpanchenko@nb1bd-03-22:~/work/arch-pc/lab06$ ./lab6-1
```

Рис. 5. Запуск файла

Этот символ не отображается при выводе на экран.

4) Создаем файл lab6-2.asm в каталоге ~/work/arch-pc/lab06 и вводим в него текст программы. (Рис. 6-7)

Создаем исполняемый файл и запускаем его. (Рис. 8)

```
ddpanchenko@nb1bd-03-22:~/work/arch-pc/lab06$ touch lab6-2.asm
```

Рис. 6. Создание файла

```
GNU nano 4.8
#include 'in_out.asm'
SECTION .text
GLOBAL _start
_start:
mov eax,'6'
mov ebx,'4'
add eax,ebx
call iprintLF
call quit
```

Рис. 7. Текст программы

```
ddpanchenko@nb1bd-03-22:~/work/arch-pc/lab06$ nasm -f elf lab6-2.asm
ddpanchenko@nb1bd-03-22:~/work/arch-pc/lab06$ ld -m elf_i386 -o lab6-2 lab6-2.o
ddpanchenko@nb1bd-03-22:~/work/arch-pc/lab06$ ./lab6-2
106
```

Рис. 8. Запуск файла

5) Аналогично предыдущему примеру изменяем символы на числа. (Рис. 9)

Создаем исполняемый файл и запускаем его. (Рис. 10)

Заменяем функцию iprintLF на iprint. Создаем исполняемый файл и запускаем его. (Рис. 11-12)

```
GNU nano 4.8
#include 'in_out.asm'
SECTION .text
GLOBAL _start
_start:
mov eax,6
mov ebx,4
add eax,ebx
call iprintLF
call quit
```

Рис. 9. Текст программы

```
ddpanchenko@nbibd-03-22:~/work/arch-pc/lab06$ nasm -f elf lab6-2.asm
ddpanchenko@nbibd-03-22:~/work/arch-pc/lab06$ ld -m elf_i386 -o lab6-2 lab6-2.o
ddpanchenko@nbibd-03-22:~/work/arch-pc/lab06$ ./lab6-2
10
```

Рис. 10. Запуск файла

```
GNU nano 4.8
#include 'in_out.asm'
SECTION .text
GLOBAL _start
_start:
mov eax,6
mov ebx,4
add eax,ebx
call iprint
call quit
```

Рис. 11. Текст программы

```
ddpanchenko@nbibd-03-22:~/work/arch-pc/lab06$ nasm -f elf lab6-2.asm
ddpanchenko@nbibd-03-22:~/work/arch-pc/lab06$ ld -m elf_i386 -o lab6-2 lab6-2.o
ddpanchenko@nbibd-03-22:~/work/arch-pc/lab06$ ./lab6-2
10ddpanchenko@nbibd-03-22:~/work/arch-pc/lab06$
```

Рис. 12. Запуск файла

Разница в том, что подпрограмма `iprintLF` переводит сообщение на следующую строку, а `iprint` данного действия не делает.

6) Создаем файл `lab6-3.asm` в каталоге `~/work/arch-pc/lab06` и вводим в него текст программы. (Рис. 13-14)

Создаем исполняемый файл и запускаем его. (Рис. 15)

Изменяем текст программы для вычисления выражения $f(x)=(4*6+2)/5$. (Рис. 16)

Создаем исполняемый файл и проверяем его работу. (Рис. 17)

```
ddpanchenko@nbibd-03-22:~/work/arch-pc/lab06$ touch lab6-3.asm
```

Рис. 13. Создание файла

```
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
```

Рис. 14. Текст программы

```
ddpanchenko@nbibd-03-22:~/work/arch-pc/lab06$ nasm -f elf lab6-3.asm
ddpanchenko@nbibd-03-22:~/work/arch-pc/lab06$ ld -m elf_i386 -o lab6-3 lab6-3.o
ddpanchenko@nbibd-03-22:~/work/arch-pc/lab06$ ./lab6-3
Результат: 4
Остаток от деления: 1
```

Рис. 15. Запуск файла

```
; ---- Вычисление выражения
mov eax,4 ; EAX=4
mov ebx,6 ; EBX=6
mul ebx ; EAX=EAX*EBX
add eax,2 ; EAX=EAX+2
xor edx,edx ; обнуляем EDX для корректной работы div
mov ebx,5 ; EBX=5
div ebx ; EAX=EAX/5, EDX=остаток от деления
```

Рис. 16. Изменения

```
ddpanchenko@nbibd-03-22:~/work/arch-pc/lab06$ nasm -f elf lab6-3.asm
ddpanchenko@nbibd-03-22:~/work/arch-pc/lab06$ ld -m elf_i386 -o lab6-3 lab6-3.o
ddpanchenko@nbibd-03-22:~/work/arch-pc/lab06$ ./lab6-3
Результат: 5
Остаток от деления: 1
```

Рис. 17. Запуск файла

7) Создаем файл variant.asm в каталоге ~/work/arch-pc/lab06 и вводим в него текст программы. (Рис. 18-19)

Создаем исполняемый файл и запускаем его. (Рис. 20)

```
ddpanchenko@nbibd-03-22:~/work/arch-pc/lab06$ touch variant.asm
```

Рис. 18. Создание файла

```
GNU nano 4.8
;-----
; Программа вычисления варианта
;-----
#include 'in_out.asm'
SECTION .data
msg: DB 'Введите Но студенческого билета: ',0
rem: DB 'Ваш вариант: ',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`
```

Рис. 19. Текст программы

```
ddpanchenko@nbtbd-03-22:~/work/arch-pc/lab06$ nasm -f elf variant.asm
ddpanchenko@nbtbd-03-22:~/work/arch-pc/lab06$ ld -m elf_i386 -o variant variant.o
ddpanchenko@nbtbd-03-22:~/work/arch-pc/lab06$ ./variant
Введите No студенческого билета:
1132229047
Ваш вариант: 8
```

Рис. 20. Запуск файла

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 – запись входной переменной в регистр ecx;
```

```
mov edx,80 – запись переменной в регистр edx;
```

```
call spread – вызов процедуры чтения данных.
```

3. Для чего используется инструкция “call atoi”?

Для вызова подпрограммы преобразования ASCII кода в число, `eax=x`.

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 ebx.

6. Для чего используется инструкция “inc edx”?

Для увеличения на 1.

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

```
mov eax,rem
```

```
call sprint
```

```
mov eax,edx
```

```
call iprintLF.
```

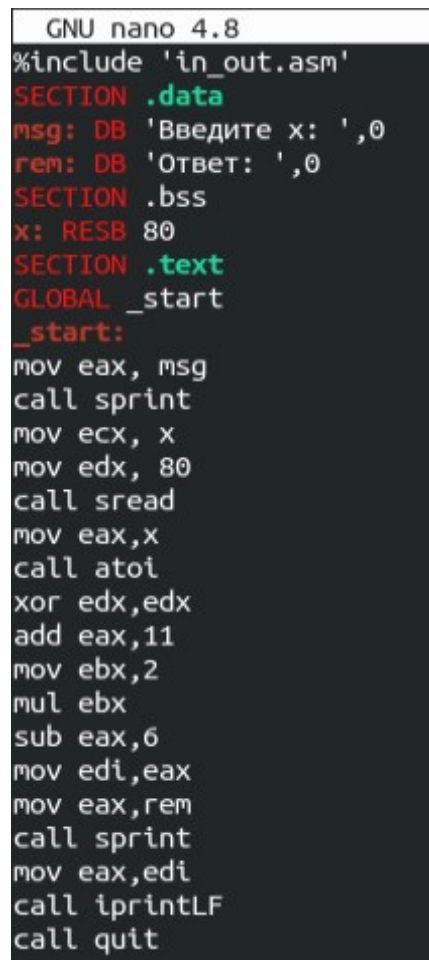

Задание для самостоятельной работы.

1) Напишем программу вычисления выражения $y = (11 + x) \cdot 2 - 6$ (Вариант 8).

(Рис. 21)

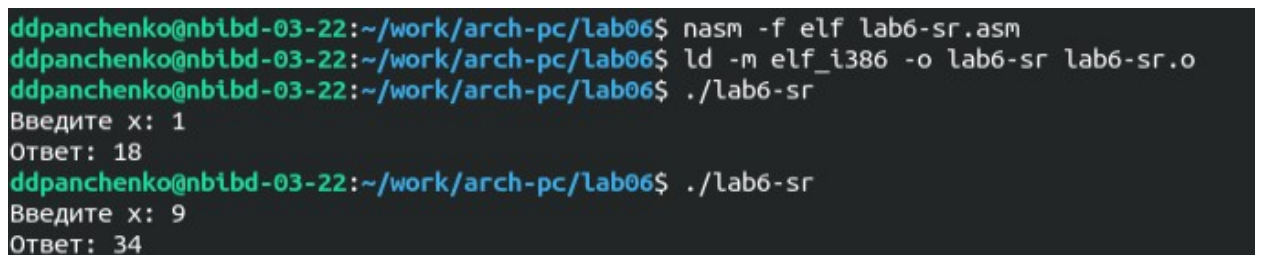
Получаем исполняемый файл и проверяем его работу для значений $x_1 = 1$ и $x_2 = 9$

(Вариант 8). (Рис. 22)



```
GNU nano 4.8
#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 sprint
mov ecx, x
mov edx, 80
call sread
mov eax, x
call atoi
xor edx, edx
add eax, 11
mov ebx, 2
mul ebx
sub eax, 6
mov edi, eax
mov eax, rem
call sprint
mov eax, edi
call iprintLF
call quit
```

Рис. 21. Текст программы



```
ddpanchenko@nbibd-03-22:~/work/arch-pc/lab06$ nasm -f elf lab6-sr.asm
ddpanchenko@nbibd-03-22:~/work/arch-pc/lab06$ ld -m elf_i386 -o lab6-sr lab6-sr.o
ddpanchenko@nbibd-03-22:~/work/arch-pc/lab06$ ./lab6-sr
Введите x: 1
Ответ: 18
ddpanchenko@nbibd-03-22:~/work/arch-pc/lab06$ ./lab6-sr
Введите x: 9
Ответ: 34
```

Рис. 22. Запуск файла

Вывод:

В ходе выполнения данной лабораторной работы я освоил арифметические инструкции языка ассемблера NASM.