

Отчёт по лабораторной работе №6

Дисциплина: Архитектура компьютера

Аветисян Алина Эдуардовна НММбд-01-23

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
	2.0.1 Ответы на вопросы.	12
3	Выполнение заданий для самостоятельной работы	14
4	Выводы	16

Список иллюстраций

2.1	Перехожу в созданный каталог с помощью утилиты cd. С помощью утилиты touch создаю файл lab6-1.asm.	6
2.2	Копирование in_out.asm.	6
2.3	Открываю файл lab6-1.asm с помощью текстового редактора nano.	6
2.4	Запуск исполняемого файла.	7
2.5	Редактирование файла.	7
2.6	Создание файла.	7
2.7	Редактирование файла.	8
2.8	Запуск исполняемого файла	8
2.9	Редактирование файла.	8
2.10	Запуск исполняемого файла.	9
2.11	Редактирование файла.	9
2.12	Запуск исполняемого файла.	9
2.13	Создание файла.	9
2.14	Редактирование файла.	10
2.15	Запуск исполняемого файла.	10
2.16	Изменение программы.	11
2.17	Запуск исполняемого файла.	11
2.18	Создание файла.	11
2.19	Редактирование файла.	12
2.20	Запуск исполняемого файла.	12
3.1	Создание файла.	14
3.2	Написание программы.	14
3.3	Запуск исполняемого файла.	15
3.4	Запуск исполняемого файла.	15

Список таблиц

1 Цель работы

Цель данной лабораторной работы - освоение арифметических инструкций языка ассемблера NASM.

2 Выполнение лабораторной работы

С помощью утилиты `mkdir` создаю директорию, в которой буду создавать файлы с программами для лабораторной работы №6.

```
aeavetisyan@user-N73SV:~$ mkdir ~/work/arch-pc/lab06
aeavetisyan@user-N73SV:~$ cd ~/work/arch-pc/lab06
aeavetisyan@user-N73SV:~/work/arch-pc/lab06$ touch lab6-1.asm
```

Рис. 2.1: Перехожу в созданный каталог с помощью утилиты `cd`. С помощью утилиты `touch` создаю файл `lab6-1.asm`.

Копирую в текущий каталог файл `in_out.asm` с помощью утилиты `cp`, т.к. он будет использоваться в других программах.

```
aeavetisyan@user-N73SV:~/work/arch-pc/lab06$ cp ~/Загрузки/in_out.asm in_out.asm
```

Рис. 2.2: Копирование `in_out.asm`.

Открываю созданный файл `lab6-1.asm`, вставляю в него программу вывода значения регистра `eax`.

```
GNU nano 6.2 /home/aeavetisyan/work/arch-pc/lab06/lab6-1.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. 2.3: Открываю файл `lab6-1.asm` с помощью текстового редактора `nano`.

Создаю исполняемый файл программы и запускаю его. Вывод программы: символ j, потому что программа вывела символ, соответствующий по системе ASCII сумме двоичных кодов символов 4 и 6.

```
aeavetisyan@user-N73SV:~/work/arch-pc/lab06$ nasm -f elf lab6-1.asm
aeavetisyan@user-N73SV:~/work/arch-pc/lab06$ ld -m elf_i386 -o lab6-1 lab6-1.o
aeavetisyan@user-N73SV:~/work/arch-pc/lab06$ ./lab6-1
j
aeavetisyan@user-N73SV:~/work/arch-pc/lab06$
```

Рис. 2.4: Запуск исполняемого файла.

Изменяю в тексте программы символы “6” и “4” на цифры 6 и 4.

```
GNU nano 6.2 /home/aeavetisyan/work/arch-pc/lab06/lab6-1.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. 2.5: Редактирование файла.

Создаю новый исполняемый файл программы и запускаю его. Теперь вывелся символ с кодом 10, это символ перевода строки, этот символ не отображается при выводе на экран.

```
aeavetisyan@user-N73SV:~/work/arch-pc/lab06$ nasm -f elf lab6-1.asm
aeavetisyan@user-N73SV:~/work/arch-pc/lab06$ ld -m elf_i386 -o lab6-1 lab6-1.o
aeavetisyan@user-N73SV:~/work/arch-pc/lab06$ ./lab6-1

aeavetisyan@user-N73SV:~/work/arch-pc/lab06$
```

{#fig:001

width=70%

Создаю новый файл lab6-2.asm с помощью утилиты touch.

```
aeavetisyan@user-N73SV:~/work/arch-pc/lab06$ touch ~/work/arch-pc/lab06/lab6-2.asm
```

Рис. 2.6: Создание файла.

Ввожу в файл текст другой программы для вывода значения регистра eax.

```
GNU nano 6.2 /home/aeavetisyan/work/arch-pc/lab06/lab6-2.asm *
#include 'in_out.asm'
SECTION .text
GLOBAL _start
_start:
mov eax,'6'
mov ebx,'4'
add eax,ebx
call iprintLF
call quit
```

Рис. 2.7: Редактирование файла.

Создаю и запускаю исполняемый файл lab6-2.asm . Теперь выводится число 106, потому что программа позволяет вывести именно число, а не символ, хотя все еще происходит именно сложение кодов символов “6” и “4”.

```
aeavetisyan@user-N73SV:~/work/arch-pc/lab06$ nasm -f elf lab6-2.asm
aeavetisyan@user-N73SV:~/work/arch-pc/lab06$ ld -m elf_i386 -o lab6-2 lab6-2.o
aeavetisyan@user-N73SV:~/work/arch-pc/lab06$ ./lab6-2
106
aeavetisyan@user-N73SV:~/work/arch-pc/lab06$
```

Рис. 2.8: Запуск исполняемого файла

Заменяю в тексте программы в файле lab6-2.asm символы “6” и “4” на числа 6 и 4.

```
GNU nano 6.2 /home/aeavetisyan/work/arch-pc/lab06/lab6-2.asm *
#include 'in_out.asm'
SECTION .text
GLOBAL _start
_start:
mov eax,6
mov ebx,4
add eax,ebx
call iprintLF
call quit
```

Рис. 2.9: Редактирование файла.

Создаю и запускаю новый исполняемый файл. Теперь программа складывает не соответствующие символам коды в системе ASCII, а сами числа, поэтому вывод 10.


```
aeavetisyan@user-N73SV:~/work/arch-pc/lab06$ nasm -f elf lab6-2.asm
aeavetisyan@user-N73SV:~/work/arch-pc/lab06$ ld -m elf_i386 -o lab6-2 lab6-2.o
aeavetisyan@user-N73SV:~/work/arch-pc/lab06$ ./lab6-2
10
aeavetisyan@user-N73SV:~/work/arch-pc/lab06$
```

Рис. 2.10: Запуск исполняемого файла.

Заменяю в тексте программы функцию `iprintLF` на `iprint`.

```
GNU nano 6.2 /home/aeavetisyan/work/arch-pc/lab06/lab6-2.asm *
#include 'in_out.asm'
SECTION .text
GLOBAL _start
_start:
mov eax,6
mov ebx,4
add eax,ebx
call iprint
call quit
```

Рис. 2.11: Редактирование файла.

Создаю и запускаю новый исполняемый файл. Вывод не изменился, потому что символ переноса строки не отображался, когда программа исполнялась с функцией `iprintLF`, а `iprint` не добавляет к выводу символ переноса строки, в отличие от `iprintLF`.

```
aeavetisyan@user-N73SV:~/work/arch-pc/lab06$ nasm -f elf lab6-2.asm
aeavetisyan@user-N73SV:~/work/arch-pc/lab06$ ld -m elf_i386 -o lab6-2 lab6-2.o
aeavetisyan@user-N73SV:~/work/arch-pc/lab06$ ./lab6-2
10aeavetisyan@user-N73SV:~/work/arch-pc/lab06$
```

Рис. 2.12: Запуск исполняемого файла.

Создаю файл `lab6-3.asm` с помощью утилиты `touch`.

```
aeavetisyan@user-N73SV:~/work/arch-pc/lab06$ touch ~/work/arch-pc/lab06/lab6-3.asm
```

Рис. 2.13: Создание файла.

Ввожу в созданный файл текст программы для вычисления значения выражения $f(x) = (5 * 2 + 3)/3$.

```

GNU nano 6.2 /home/aeavetisyan/work/arch-pc/lab06/lab6-3.asm *
;-----
; Программа вычисления выражения
;-----
#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 iprintf ; 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 iprintf ; из 'edx' (остаток) в виде символов
call quit ; вызов подпрограммы завершения

```

Рис. 2.14: Редактирование файла.

Создаю исполняемый файл и запускаю его.

```

aeavetisyan@user-N73SV:~/work/arch-pc/lab06$ nasm -f elf lab6-3.asm
aeavetisyan@user-N73SV:~/work/arch-pc/lab06$ ld -m elf_i386 -o lab6-3 lab6-3.o
aeavetisyan@user-N73SV:~/work/arch-pc/lab06$ ./lab6-3
Результат: 4
Остаток от деления: 1
aeavetisyan@user-N73SV:~/work/arch-pc/lab06$

```

Рис. 2.15: Запуск исполняемого файла.

Изменяю программу так, чтобы она вычисляла значение выражения $f(x) = (4 * 6 + 2)/5$.

```

GNU nano 6.2 /home/aeavetisyan/work/arch-pc/lab06/lab6-3.asm
;-----
; Программа вычисления выражения
;-----
#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=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
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. 2.16: Изменение программы.

Создаю и запускаю новый исполняемый файл.

```

aeavetisyan@user-N735V:~/work/arch-pc/lab06$ nasm -f elf lab6-3.asm
aeavetisyan@user-N735V:~/work/arch-pc/lab06$ ld -m elf_i386 -o lab6-3 lab6-3.o
aeavetisyan@user-N735V:~/work/arch-pc/lab06$ ./lab6-3
Результат: 5
Остаток от деления: 1
aeavetisyan@user-N735V:~/work/arch-pc/lab06$

```

Рис. 2.17: Запуск исполняемого файла.

Создаю файл variant.asm с помощью утилиты touch.

```

aeavetisyan@user-N735V:~/work/arch-pc/lab06$ touch ~/work/arch-pc/lab06/variant.asm

```

Рис. 2.18: Создание файла.

Ввожу в файл текст программы для вычисления варианта задания по номеру студенческого билета.

```
GNU nano 6.2 /home/aeavetisyan/work/arch-pc/lab06/variant.asm *
;-----
; Программа вычисления варианта
;-----
#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 ; вызов подпрограммы преобразования
call atoi ; ASCII кода в число, 'eax=x'
xor edx, edx
mov ebx, 20
div ebx
inc edx
mov eax, rem
call sprint
mov eax, edx
call iprintLF
call quit
█
```

Рис. 2.19: Редактирование файла.

Создаю и запускаю исполняемый файл. Ввожу номер своего студенческого билета с клавиатуры, программа вывела, что мой вариант 15.

```
aeavetisyan@user-N73SV:~/work/arch-pc/lab06$ nasm -f elf variant.asm
aeavetisyan@user-N73SV:~/work/arch-pc/lab06$ ld -m elf_i386 -o variant variant.o
aeavetisyan@user-N73SV:~/work/arch-pc/lab06$ ./variant
Введите № студенческого билета:
1132236074
Ваш вариант: 15
aeavetisyan@user-N73SV:~/work/arch-pc/lab06$ █
```

Рис. 2.20: Запуск исполняемого файла.

2.0.1 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 `div` `mov 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 Выполнение заданий для самостоятельной работы

Создаю файл lab6-4.asm с помощью утилиты touch.

```
aeavetisyan@user-N73SV:~/work/arch-pc/lab06$ touch ~/work/arch-pc/lab06/lab6-4.asm
aeavetisyan@user-N73SV:~/work/arch-pc/lab06$
```

Рис. 3.1: Создание файла.

Открываю созданный файл для редактирования, ввожу в него текст программы для вычисления значения выражения $f(x) = (5 + x)^2 - 3$. Это выражение было под вариантом 15.

```
GNU nano 6.2 /home/aeavetisyan/work/arch-pc/lab06/lab6-4.asm
#include "in_out.asm" ; подключение внешнего файла
SECTION .data ; секция инициализированных данных
msg: DB "Введите значение переменной x: ",0
res: DB "Результат: ",0
SECTION .bss ; секция не инициализированных данных
x: RESB 80 ; Переменная, значение к-рой будем вводить с клавиатуры, выделенный размер - 80 байт
SECTION .text ; Код программы
GLOBAL _start ; Начало программы
_start: ; Точка входа в программу
; --- Вычисление выражения
mov eax, msg ; запись адреса выводимого сообщения в eax
call sprint ; вызов подпрограммы печати сообщения
mov ecx, x ; запись адреса переменной в ecx
mov edx, 80 ; запись длины вводимого значения в edx
call sread ; вызов подпрограммы ввода сообщения
mov eax, x ; вызов подпрограммы преобразования
call atoi ; ASCII кода в число, 'eax=x'
add eax, 5 ; EAX = EAX + 5 = x + 5
mov ebx, eax ; запись значения x в регистр ebx
mul ebx ; EAX=EAX*EBX = (x+5)*(x+5)
;call atoi ; ASCII кода в число, 'EBX=x'
;add ebx, 5 ; EBX = EBX + 5 = x + 5
;mul ebx ; EAX=EAX*EBX = (x+5)*(x+5)
sub eax, 3 ; EAX = EAX - 3 = (x+5)*(x+5) - 3
mov edi, eax ; запись результата вычисления в 'edi'
; --- Вывод результата на экран
mov eax, res ; вызов подпрограммы печати
call sprint ; сообщения 'Результат: '
mov eax, edi ; вызов подпрограммы печати значения
call lprint ; из 'edi' в виде символов
call quit ; вызов подпрограммы завершения
```

Рис. 3.2: Написание программы.

Создаю и запускаю исполняемый файл. При вводе значения 5, вывод 97.

```

aeavetisyan@user-N73SV:~/work/arch-pc/lab06$ nasm -f elf lab6-4.asm
aeavetisyan@user-N73SV:~/work/arch-pc/lab06$ ld -m elf_i386 -o lab6-4 lab6-4.o
aeavetisyan@user-N73SV:~/work/arch-pc/lab06$ ./lab6-4
Введите значение переменной x: 5
Результат: 97aeavetisyan@user-N73SV:~/work/arch-pc/lab06$

```

Рис. 3.3: Запуск исполняемого файла.

Провожу еще один запуск исполняемого файла для проверки работы программы с другим значением на входе. Программа отработала верно. При вводе значения 1, вывод 33.

```

Результат: 97aeavetisyan@user-N73SV:~/work/arch-pc/lab06$ nasm -f elf lab6-4.asm
aeavetisyan@user-N73SV:~/work/arch-pc/lab06$ ld -m elf_i386 -o lab6-4 lab6-4.o
aeavetisyan@user-N73SV:~/work/arch-pc/lab06$ ./lab6-4
Введите значение переменной x: 1
Результат: 33aeavetisyan@user-N73SV:~/work/arch-pc/lab06$

```

Рис. 3.4: Запуск исполняемого файла.

4 Выводы

При выполнении данной лабораторной работы я освоила арифметические инструкции языка ассемблера NASM.