

# **Отчёт по лабораторной работе №7**

**Дисциплина: архитектура компьютера**

Сидельников Андрей Владимирович

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Выполнение лабораторной работы</b>	<b>6</b>
2.1	Самостоятельная работа . . . . .	8
<b>3</b>	<b>Выводы</b>	<b>12</b>

## **Список таблиц**

# Список иллюстраций

figno:1. . . . .	6
2.1 Создание файла и работа программы . . . . .	6
2.2 Работа программы . . . . .	6
2.3 Результат программы . . . . .	6
figno:2. Сам файл . . . . .	7
2.4 Проверка программы . . . . .	7
2.5 Создание файла . . . . .	7
2.6 Строки кода . . . . .	8
figno:3. Ошибка трансляции lab7-2 . . . . .	8
2.7 Файл листинга с ошибкой lab7-2 . . . . .	8
figno:4. Программа . . . . .	9
2.8 Проверка программы . . . . .	9
figno:5. Программа в файле lab7-4.asm . . . . .	10
2.9 Запуск программы lab7-4.asm . . . . .	11

# 1 Цель работы

Целью работы является изучение команд условного и безусловного переходов. Приобретение навыков написания программ с использованием переходов. Знакомство с назначением и структурой файла листинга.

## 2 Выполнение лабораторной работы

Создаю файл lab7-1.asm , предварительно копирую в него файл in\_out.asm и вписываю в него код из листинга 7.1. (рис. 2.1)

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

avsideljnikov@avsideljnikov-VirtualBox:~$ cp /home/avsideljnikov/Записки/in_out.asm /home/avsideljnikov/
work/arch-pc/lab07
avsideljnikov@avsideljnikov-VirtualBox:~$ cd ~/work/arch-pc/lab07
avsideljnikov@avsideljnikov-VirtualBox:~/work/arch-pc/lab07$ nasm -f elf lab7-1.asm
avsideljnikov@avsideljnikov-VirtualBox:~/work/arch-pc/lab07$ ld -m elf_i386 -o lab7-1 lab7-1.o
avsideljnikov@avsideljnikov-VirtualBox:~/work/arch-pc/lab07$ ./lab7-1
Сообщение № 2
Сообщение № 3
avsideljnikov@avsideljnikov-VirtualBox:~/work/arch-pc/lab07$
```

Рис. 2.1: Создание файла и работа программы

Запишем код программы из листинга 7.2 и также проверим его работу (рис. 2.2)

```
avsideljnikov@avsideljnikov-VirtualBox:~/work/arch-pc/lab07$ nasm -f elf lab7-1.asm
avsideljnikov@avsideljnikov-VirtualBox:~/work/arch-pc/lab07$ ld -m elf_i386 -o lab7-1 lab7-1.o
avsideljnikov@avsideljnikov-VirtualBox:~/work/arch-pc/lab07$ ./lab7-1
Сообщение № 2
Сообщение № 3
avsideljnikov@avsideljnikov-VirtualBox:~/work/arch-pc/lab07$
```

Рис. 2.2: Работа программы

Изменив текст программы ,получаем интересный нас результат (рис. 2.3)

```
avsideljnikov@avsideljnikov-VirtualBox:~/work/arch-pc/lab07$ nasm -f elf lab7-1.asm
avsideljnikov@avsideljnikov-VirtualBox:~/work/arch-pc/lab07$ ld -m elf_i386 -o lab7-1 lab7-1.o
avsideljnikov@avsideljnikov-VirtualBox:~/work/arch-pc/lab07$ ./lab7-1
Сообщение № 3
Сообщение № 2
Сообщение № 1
```

Рис. 2.3: Результат программы

Создание файла lab7-2.asm. и проверка программы из листинга 7.3 (рис. 2.4)

```
avsideljnikov@avsideljnikov-VirtualBox:~/work/arch-pc/lab07$ ./lab7-2
Введите В: 9
Наибольшее число: 50
```

Сам файл

```
avsideljnikov@avsideljnikov-VirtualBox:~/work/arch-pc/lab07$ touch lab7-2.asm
avsideljnikov@avsideljnikov-VirtualBox:~/work/arch-pc/lab07$ nasm -f elf lab7-2.asm
avsideljnikov@avsideljnikov-VirtualBox:~/work/arch-pc/lab07$ ld -m elf_i386 -o lab7-2 lab7-2.o
avsideljnikov@avsideljnikov-VirtualBox:~/work/arch-pc/lab07$ ./lab7-2
Введите В: 100
Наибольшее число: 100
```

Рис. 2.4: Проверка программы

Создаю файл листинг и открываю его (рис. 2.5)

lab7-2.lst		
1	1	%include 'in_out.asm'
2	2	<1> ;----- slen -----
3	3	<1> ; Функция вычисления длины сообщения
4	4	<1> slen:
5	5	<1> push ebx
6	6	<1> mov ebx, eax
7	7	<1>
8	8	<1> nextchar:
9	9	<1> cmp byte [eax], 0
10	10	<1> jz finished
11	11	<1> inc eax
12	12	<1> jmp nextchar
13	13	<1>
14	14	<1> finished:
15	15	<1> sub eax, ebx
16	16	<1> pop ebx
17	17	<1> ret
18	18	<1>
19	19	<1>
20	20	<1> ;----- sprint -----
21	21	<1> ; Функция печати сообщения
22	22	<1> ; входные данные: mov eax,<message>
23	23	<1> sprint:
24	24	<1> push edx
25	25	<1> push ecx
26	26	<1> push ebx
27	27	<1> push eax
28	28	<1> call slen
29	29	<1>
30	30	<1> mov edx, eax
31	31	<1> pop eax
32	32	<1>
33	33	<1> mov ecx, eax
34	34	<1> mov ebx, 1
35	35	<1> mov eax, 4
36	36	<1> int 80h
37	37	<1>
38	38	<1> pop ebx
39	39	<1> pop ecx
40	40	<1> pop edx
41	41	<1> ret
42	42	<1>
43	43	<1>

Рис. 2.5: Создание файла

В строке 193 находится номер строки в подпрограмме - 18 , адрес - 000000F7 ,

машинный код - BA0A000000 и сам исходный код - mov edx,10 - перемещение 10 в edx

В строке 189 находится номер строки в подпрограмме - 14 , адрес - 000000E8, машинный код - B8[00000000] и сам исходный код - mov eax,msg1 - перемещение msg1 в eax

В строке 192 находится номер строки в подпрограмме - 17 , адрес - 000000F2 , машинный код - B9[0A000000] и сам исходный код - mov ecx,B - перемещение B в ecx (рис. 2.6)

189	14	000000E8	B8[00000000]	mov eax,msg1	
190	15	000000ED	E81DFFFFFF	call sprint	
191	16			; ----- Ввод 'B'	
192	17	000000F2	B9[0A000000]	mov ecx,B	
193	18	000000F7	BA0A000000	mov edx,10	2
194	19	000000FC	E842FFFFFF	call sread	
195	20			; ----- Преобразование 'B' из символа в	

Рис. 2.6: Строки кода

Открыл файл с программой lab7-2.asm и в инструкции с двумя операндами удалил один операнд. Выполнил трансляцию с получением файла листинга (рис. 2.7)

```
avsideljnikov@avsideljnikov-VirtualBox:~/work/arch-pc/lab07$ nasm -f elf -l lab7-2.lst lab7-2.asm
lab7-2.asm:34: error: invalid combination of opcode and operands
avsideljnikov@avsideljnikov-VirtualBox:~/work/arch-pc/lab07$
```

Ошибка трансляции lab7-2

207	32			; ----- преобразование max(A,C) из	
			символа в число		
208	33			check_B:	2
209	34			mov eax,	2
210	34		*****	error: invalid combination of opcode and	
			operands]		
211	35	00000130	E867FFFFFF	call atoi ; Вызов подпрограммы перевода	

Рис. 2.7: Файл листинга с ошибкой lab7-2

Объектный файл не смог создаться из-за ошибки, но получился листинг, где выделено место ошибки.

## 2.1 Самостоятельная работа

Так как у меня 6 вариант , то мои целые числа будут 79,83,41 (рис. 2.8)



```

lab7-3.asm
1 %include 'in_out.asm'
2 SECTION .data
3     msgA:      DB 'Введите A: ',0
4     msgB:      DB 'Введите B: ',0
5     msgC:      DB 'Введите C: ',0
6     answer:    DB 'Наименьшее: ',0
7
8 SECTION .bss
9     A:  RESB 80
10    B:  RESB 80
11    C:  RESB 80
12    result:  RESB 80
13    min: RESB 80
14
15 SECTION .text
16     GLOBAL _start
17
18 _start:
19     mov eax,msgA
20     call sprint
21     mov ecx,A
22     mov edx,80
23     call sread
24     mov eax,A
25     call atoi
26     mov [A],eax
27
28     mov eax, msgB
29     call sprint
30     mov ecx,B
31     mov edx,80
32     call sread
33     mov eax,B
34     call atoi
35     mov [B],eax
36
37     mov eax,msgC
38     call sprint
39     mov ecx,C
40     mov edx,80
41     call sread

```

Программа

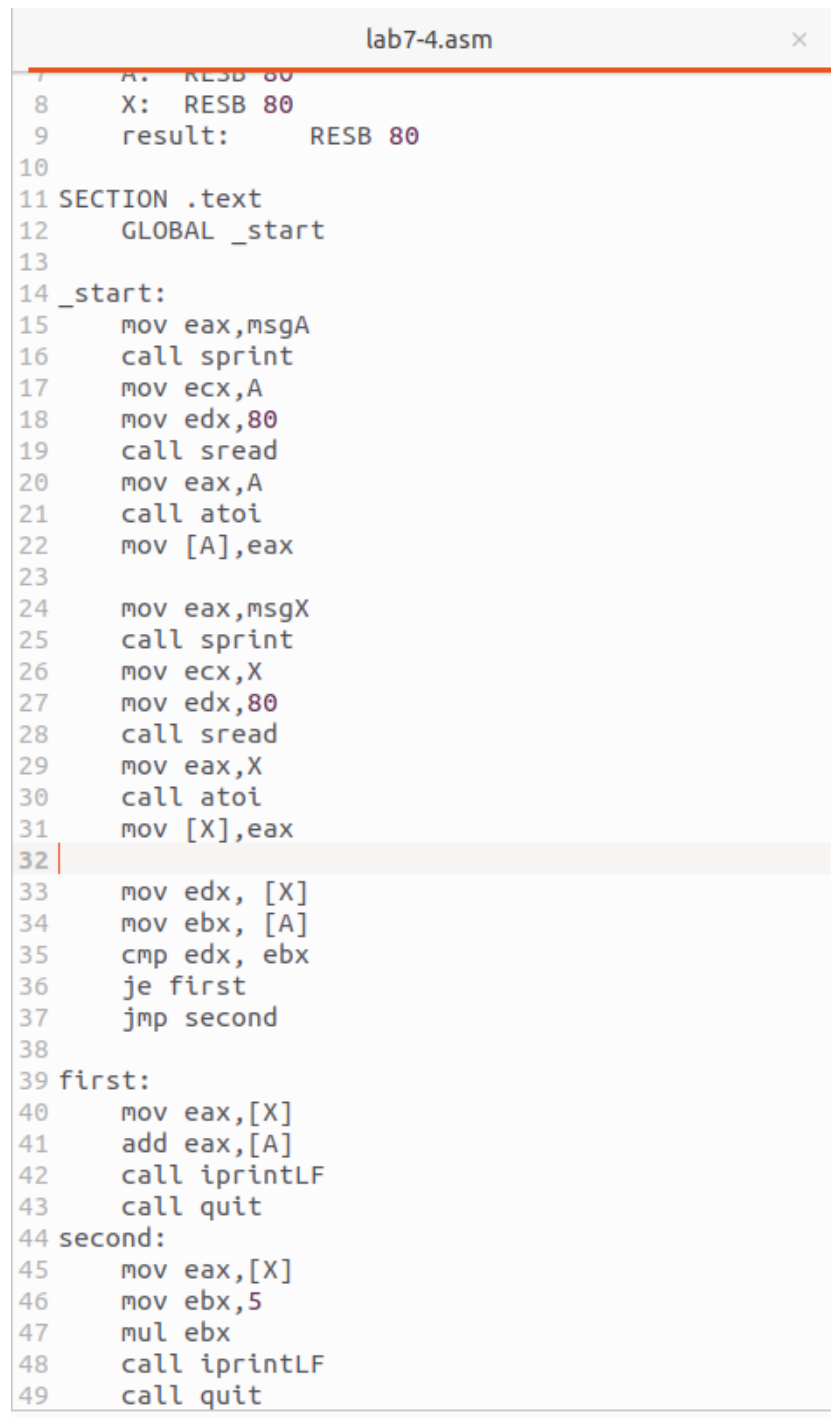
```

avsideljnikov@avsideljnikov-VirtualBox:~/work/arch-pc/lab07$ ./lab7-3
Введите A: 79
Введите B: 83
Введите C: 41
Наименьшее: 41
avsideljnikov@avsideljnikov-VirtualBox:~/work/arch-pc/lab07$

```

Рис. 2.8: Проверка программы

6 вариант предлагает систему из двух уравнений,  $x+a$ ,  $x=a$  и  $5x$ ,  $x \neq a$ . Напишем для неё программу и проверим её (рис. 2.9)



```
lab7-4.asm
7  A:  RESB 80
8  X:  RESB 80
9  result:  RESB 80
10
11 SECTION .text
12     GLOBAL _start
13
14 _start:
15     mov eax,msgA
16     call sprint
17     mov ecx,A
18     mov edx,80
19     call sread
20     mov eax,A
21     call atoi
22     mov [A],eax
23
24     mov eax,msgX
25     call sprint
26     mov ecx,X
27     mov edx,80
28     call sread
29     mov eax,X
30     call atoi
31     mov [X],eax
32
33     mov edx, [X]
34     mov ebx, [A]
35     cmp edx, ebx
36     je first
37     jmp second
38
39 first:
40     mov eax,[X]
41     add eax,[A]
42     call iprintLF
43     call quit
44 second:
45     mov eax,[X]
46     mov ebx,5
47     mul ebx
48     call iprintLF
49     call quit
```

Программы в файле lab7-4.asm

```
19/12/2007
avsideljnikov@avsideljnikov-VirtualBox:~/work/arch-pc/lab07$ nasm -f elf lab7-4.asm
avsideljnikov@avsideljnikov-VirtualBox:~/work/arch-pc/lab07$ ld -m elf_i386 -o lab7-4 lab7-4.o
avsideljnikov@avsideljnikov-VirtualBox:~/work/arch-pc/lab07$ ./lab7-4
Введите A: 2
Введите X: 2
4
avsideljnikov@avsideljnikov-VirtualBox:~/work/arch-pc/lab07$ ./lab7-4
Введите A: 1
Введите X: 2
10
avsideljnikov@avsideljnikov-VirtualBox:~/work/arch-pc/lab07$
```

Рис. 2.9: Запуск программы lab7-4.asm

## 3 Выводы

Я изучил команды условного и безусловного переходов и научился писать программы с использованием этих переходов.