

# РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра прикладной информатики и теории вероятностей

## ОТЧЕТ

### ПО ЛАБОРАТОРНОЙ РАБОТЕ № 7

дисциплина:     Архитектура компьютера

Студент: Кондрацкая Александра Евгеньевна

Группа: НММ-02-24

МОСКВА

2024 г.

## Оглавление

1. Цель работы.....	3
2. Задание.....	4
3. Выполнение лабораторной работы.....	5
4. Выводы.....	15

## **1. Цель работы**

Цель заключается в изучении команд условного и безусловного переходов, а также в приобретении навыков написания программ с их использованием. Кроме того, планируется ознакомиться с назначением и структурой файла листинга.

## **2. Задание**

1. Реализация переходов в NASM
2. Изучение структуры файлы листинга

### 3. Выполнение лабораторной работы

Создадим каталог для программ лабораторной работы № 7, перейдём в него и создадим файл lab7-1.asm (рис.1)

```
aekondrackaya@dk3n55 ~ $ mkdir ~/work/arch-pc/lab07
aekondrackaya@dk3n55 ~ $ cd ~/work/arch-pc/lab07
aekondrackaya@dk3n55 ~/work/arch-pc/lab07 $ touch lab7-1.asm
aekondrackaya@dk3n55 ~/work/arch-pc/lab07 $ ls
lab7-1.asm
aekondrackaya@dk3n55 ~/work/arch-pc/lab07 $
```

Рис.1 Создание папки и файла

Введите в файл lab7-1.asm текст программы из листинга 7.1. (рис. 2)

```
1 %include 'in_out.asm' ; подключение внешнего файла
2 SECTION .data
3 msg1: DB 'Сообщение № 1',0
4 msg2: DB 'Сообщение № 2',0
5 msg3: DB 'Сообщение № 3',0
6 SECTION .text
7 GLOBAL _start
8 _start:
9 jmp _label2
10 _label1:
11 mov eax, msg1 ; Вывод на экран строки
12 call sprintfLF ; 'Сообщение № 1'
13 _label2:
14 mov eax, msg2 ; Вывод на экран строки
15 call sprintfLF ; 'Сообщение № 2'
16 _label3:
17 mov eax, msg3 ; Вывод на экран строки
18 call sprintfLF ; 'Сообщение № 3'
19 _end:
20 call quit ; вызов подпрограммы завершения
```

Рис. 2 ввод программы

Создадим исполняемый файл и запустим его (рис. 3)

```
aecondrackaya@dk3n55 ~/work/arch-pc/lab07 $ nasm -f elf lab7-1.asm
aecondrackaya@dk3n55 ~/work/arch-pc/lab07 $ ld -m elf_i386 -o lab7-1 lab7-1.o
aecondrackaya@dk3n55 ~/work/arch-pc/lab07 $ ./lab6-2
bash: ./lab6-2: Нет такого файла или каталога
aecondrackaya@dk3n55 ~/work/arch-pc/lab07 $ ./lab7-1
Сообщение № 2
Сообщение № 3
aecondrackaya@dk3n55 ~/work/arch-pc/lab07 $
```

Рис. 3 Создание исполняемого файла

Изменим текст программы в соответствии с листингом 7.2. (рис. 4)

```
1 SECTION .data
2 msg1: DB 'Сообщение № 1',0
3 msg2: DB 'Сообщение № 2',0
4 msg3: DB 'Сообщение № 3',0
5 SECTION .text
6 GLOBAL _start
7 _start:
8 jmp _label2
9 _label1:
10 mov eax, msg1 ; Вывод на экран строки
11 call sprintf ; 'Сообщение № 1'
12 jmp _end
13 _label2:
14 mov eax, msg2 ; Вывод на экран строки
15 call sprintf ; 'Сообщение № 2'
16 jmp _label1
17 _label3:
18 mov eax, msg3 ; Вывод на экран строки
19 call sprintf ; 'Сообщение № 3'
20 _end:
21 call quit ; вызов подпрограммы завершения
```

Рис. 4 Изменение программы

Изменим текст программы добавив или изменив инструкции jmp (рис. 5)

```

1 %include 'in_out.asm' ; подключение внешнего файла
2 SECTION .data
3 msg1: DB 'Сообщение № 1',0
4 msg2: DB 'Сообщение № 2',0
5 msg3: DB 'Сообщение № 3',0
6 SECTION .text
7 GLOBAL _start
8 _start:
9 jmp _label3
10 _label1:
11 mov eax, msg1 ; Вывод на экран строки
12 call sprintf ; 'Сообщение № 1'
13 jmp _end
14 _label2:
15 mov eax, msg2 ; Вывод на экран строки
16 call sprintf ; 'Сообщение № 2'
17 jmp _label1
18 _label3:
19 mov eax, msg3 ; Вывод на экран строки
20 call sprintf ; 'Сообщение № 3'
21 jmp _label2
22 _end:
23 call quit ; вызов подпрограммы завершения

```

Рис.5 изменение программы

Создадим исполняемый файл и проверим его работу. (рис. 6)

```

aekondrackaya@dk3n55 ~/work/arch-pc/lab07 $ gedit lab7-1.asm
aekondrackaya@dk3n55 ~/work/arch-pc/lab07 $ nasm -f elf lab7-1.asm
aekondrackaya@dk3n55 ~/work/arch-pc/lab07 $ ld -m elf_i386 -o lab7-1 lab7-1.o
aekondrackaya@dk3n55 ~/work/arch-pc/lab07 $ ./lab7-1
Сообщение № 2
Сообщение № 1
aekondrackaya@dk3n55 ~/work/arch-pc/lab07 $ gedit lab7-1.asm
aekondrackaya@dk3n55 ~/work/arch-pc/lab07 $ nasm -f elf lab7-1.asm
aekondrackaya@dk3n55 ~/work/arch-pc/lab07 $ ld -m elf_i386 -o lab7-1 lab7-1.o
aekondrackaya@dk3n55 ~/work/arch-pc/lab07 $ ./lab7-1
Сообщение № 3
Сообщение № 2
Сообщение № 1
aekondrackaya@dk3n55 ~/work/arch-pc/lab07 $ gedit lab7-1.asm
aekondrackaya@dk3n55 ~/work/arch-pc/lab07 $

```

Рис. 6 Создание исполняемого файла до и после изменений

Создадим файл lab7-2.asm в каталоге ~/work/arch-pc/lab07. Внимательно изучим текст программы из листинга 7.3 и введём в lab7-2.asm. (рис. 7 и 8)

```

aekondrackaya@dk3n55 ~/work/arch-pc/lab07 $ touch lab7-2.asm
aekondrackaya@dk3n55 ~/work/arch-pc/lab07 $ ls
in_out.asm  lab7-1  lab7-1.asm  lab7-1.o  lab7-2.asm
aekondrackaya@dk3n55 ~/work/arch-pc/lab07 $

```

```

Открыть ▼ + *lab7-2.asm
~/work/arch-pc/lab07

1 %include 'in_out.asm'
2 section .data
3 msg1 db 'Введите B: ',0h
4 msg2 db "Наибольшее число: ",0h
5 A dd '20'
6 C dd '50'
7 section .bss
8 max resb 10
9 B resb 10
10 section .text
11 global _start
12 _start:
13 ; ----- Вывод сообщения 'Введите B: '
14 mov eax,msg1
15 call sprint
16 ; ----- Ввод 'B'
17 mov ecx,B
18 mov edx,10
19 call sread
20 ; ----- Преобразование 'B' из символа в число
21 mov eax,B
22 call atoi ; Вызов подпрограммы перевода символа в число
23 mov [B],eax ; запись преобразованного числа в 'B'

```



Рис. 8 Ввод программы

Создадим исполняемый файл и проверим его работу для разных значений В. (рис. 9)

```

aekondrackaya@dk3n55 ~/work/arch-pc/lab07 $ nasm -f elf lab7-2.asm
aekondrackaya@dk3n55 ~/work/arch-pc/lab07 $ ld -m elf_i386 -o lab7-2 lab7-2.o
aekondrackaya@dk3n55 ~/work/arch-pc/lab07 $ ./lab7-2
Введите В: 5
Наибольшее число: 50
aekondrackaya@dk3n55 ~/work/arch-pc/lab07 $ nasm -f elf lab7-2.asm
aekondrackaya@dk3n55 ~/work/arch-pc/lab07 $ ld -m elf_i386 -o lab7-2 lab7-2.o
aekondrackaya@dk3n55 ~/work/arch-pc/lab07 $ ./lab7-2
Введите В: 9
Наибольшее число: 50
aekondrackaya@dk3n55 ~/work/arch-pc/lab07 $
```

Рис. 9 Создание исполняемого файла

Создадим файл листинга для программы из файла lab7-2.asm (рис. 10)

```

aekondrackaya@dk3n55 ~/work/arch-pc/lab07 $ nasm -f elf -l lab7-2.lst lab7-2.asm
```

Рис. 10 Создаем файл листинга

В строчке 28 уберём операнд, из-за чего при создании файла появиться ошибка (рис. 11 и 12 и 13)

```

28 cmp ecx, ; Сравниваем 'A' и 'C'
29 if check B : если 'A>C' то переход на:
```

Рис. 11 Убрали операнд

```

aekondrackaya@dk3n55 ~/work/arch-pc/lab07 $ nasm -f elf -l lab7-2.lst lab7-2.asm
lab7-2.asm:28: error: invalid combination of opcode and operands
```

Рис. 12 Обнаружение ошибки

```

28 *****
29 00000116 7506
```

Рис. 13 Ошибка в листинге

## САМОСТОЯТЕЛЬНАЯ РАБОТА

### вариант 13

#### №1

Создадим файл (рис. 14)

```
aecondrackaya@dk3n55 ~/work/arch-pc/lab07 $ touch lab7-3.asm
aecondrackaya@dk3n55 ~/work/arch-pc/lab07 $ ls
in_out.asm lab7-1 lab7-1.asm lab7-1.o lab7-2 lab7-2.asm lab7-2.lst lab7-3.asm
aecondrackaya@dk3n55 ~/work/arch-pc/lab07 $
```

Рис. 14 Создание файла

Введем программу (рис. 15-17)

```
1 %include 'in_out.asm'
2
3 SECTION .data
4 A1 DB 'Введите число A: ',0h
5 B1 DB 'Введите число B: ',0h
6 C1 DB 'Введите число C: ',0h
7 otv DB 'Наименьшее число: ',0h
8 SECTION .bss
9 min RESB 20
10 A RESB 20
11 B RESB 20
12 C RESB 20
13
14 SECTION .text
15 GLOBAL _start
16 _start:
17
18 mov eax,A1
19 call sprint
20
21 mov ecx,A
22 mov edx,20
23 call sread
24
25 mov eax, A
26 call atoi
27 mov [A],eax
28
29 xor eax,eax
30
31 mov eax,B1
32 call sprint
```

Рис.15 Ввод программы

```

33
34 mov ecx,B
35 mov edx,20
36 call sread
37
38 mov eax,B
39 call atoi
40 mov [B],eax
41
42 xor eax,eax
43
44 mov ecx, [A]
45 mov [min],ecx
46 mov ecx,[min]
47
48 cmp ecx,[B]
49 jl check_C
50 mov ecx, [B]
51 mov [min],ecx
52
53 check_C:
54
55 mov eax,C1
56 call sprint
57
58 mov ecx,C
59 mov edx,10
60 call sread
61
62 mov eax,C
63 call atoi

```

---

Рис. 16 Ввод программы

```

64 mov [C],eax
65
66 xor eax,eax
67
68
69 mov ecx,[min]
70 cmp ecx,[C]
71 jl fin
72 mov ecx,[C]
73 mov [min],ecx
74
75
76 fin:
77 mov eax, otv
78 call sprint
79 mov eax, [min]
80 call iprintLF
81 call quit

```

---

Рис. 17 Ввод программы

Проверим (рис. 18)

```
aecondrackaya@dk3n55 ~/work/arch-pc/lab07 $ nasm -f elf lab7-3.asm
aecondrackaya@dk3n55 ~/work/arch-pc/lab07 $ ld -m elf_i386 -o lab7-3 lab7-3.o
aecondrackaya@dk3n55 ~/work/arch-pc/lab07 $ ./lab7-3
Введите число A: 84
Введите число B: 32
Введите число C: 77
Наименьшее число: 32
aecondrackaya@dk3n55 ~/work/arch-pc/lab07 $
```

Рис.18 Проверка

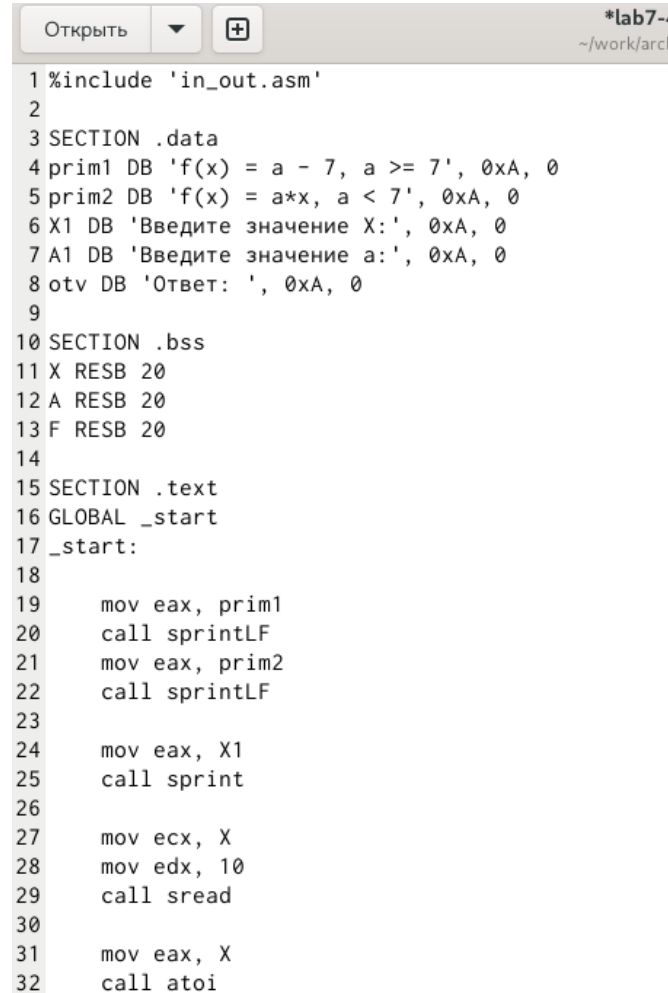
№2

Создадим файл (рис. 19)

```
aecondrackaya@dk3n55 ~/work/arch-pc/lab07 $ touch lab7-4.asm
aecondrackaya@dk3n55 ~/work/arch-pc/lab07 $ ls
in_out.asm  lab7-1.asm  lab7-2      lab7-2.lst  lab7-3.asm  lab7-4.asm
lab7-1      lab7-1.o    lab7-2.asm  lab7-3     lab7-3.o
aecondrackaya@dk3n55 ~/work/arch-pc/lab07 $
```

Рис. 19 Создание файла

Введем программу (рис. 20-22)



```
*lab7-4.asm
~/work/arch-pc/lab07

1 %include 'in_out.asm'
2
3 SECTION .data
4 prim1 DB 'f(x) = a - 7, a >= 7', 0xA, 0
5 prim2 DB 'f(x) = a*x, a < 7', 0xA, 0
6 X1 DB 'Введите значение X:', 0xA, 0
7 A1 DB 'Введите значение a:', 0xA, 0
8 otv DB 'Ответ: ', 0xA, 0
9
10 SECTION .bss
11 X RESB 20
12 A RESB 20
13 F RESB 20
14
15 SECTION .text
16 GLOBAL _start
17 _start:
18
19     mov eax, prim1
20     call sprintf
21     mov eax, prim2
22     call sprintf
23
24     mov eax, X1
25     call sprintf
26
27     mov ecx, X
28     mov edx, 10
29     call sread
30
31     mov eax, X
32     call atoi
```

Рис. 20 Ввод программы

```

33     mov [X], eax
34
35     mov eax, A1
36     call sprint
37
38     mov ecx, A
39     mov edx, 10
40     call sread
41
42     mov eax, A
43     call atoi
44     mov [A], eax
45
46     mov eax, [A]
47     cmp eax, 7
48     jl less_than_7
49
50     ; a >= 7: f(x) = a - 7
51     sub eax, 7
52     mov [F], eax
53     jmp fin
54
55 less_than_7:
56     ; a < 7: f(x) = a * x
57     mov ebx, [X]
58     imul ebx    ; eax = eax * ebx (умножение eax на ebx)
59     mov [F], eax
60
61 fin:
62     mov eax, otv

```

Рис. 21 Ввод программы

```

62     mov eax, otv
63     call sprint
64     mov eax, [F]
65     call iprintLF
66     call quit

```

Рис. 22 Ввод программы

Проверим программу (рис. 23)

```

aekondrackaya@dk3n55 ~/work/arch-pc/lab07 $ nasm -f elf lab7-4.asm
aekondrackaya@dk3n55 ~/work/arch-pc/lab07 $ ld -m elf_i386 -o lab7-4 lab7-4.o
aekondrackaya@dk3n55 ~/work/arch-pc/lab07 $ ./lab7-4
f(x) = a - 7, a >= 7

f(x) = a*x, a < 7

Введите значение X:
3
Введите значение a:
9
Ответ:
2
aekondrackaya@dk3n55 ~/work/arch-pc/lab07 $ ./lab7-4
f(x) = a - 7, a >= 7

f(x) = a*x, a < 7

Введите значение X:
6
Введите значение a:
4
Ответ:
24
aekondrackaya@dk3n55 ~/work/arch-pc/lab07 $ █

```

Рис. 23 Проверка

## **4. Выводы**

В результате работы были изучены команды условного и безусловного переходов, приобретены навыки написания программ с их использованием, а также ознакомились с назначением и структурой файла листинга.