

# **Лабораторная работа №8**

**Дисциплина: Архитектура компьютера**

Ким Ангелина Павловна

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Выполнение лабораторной работы</b>	<b>6</b>
<b>3</b>	<b>Выводы</b>	<b>23</b>
	<b>Список литературы</b>	<b>24</b>

## Список иллюстраций

2.1	Создание каталога и файла . . . . .	6
2.2	Текст программы из листинга 8.1 . . . . .	7
2.3	Создание исполняемого файла . . . . .	8
2.4	Изменяем текст программы в соответствии с листингом 8.2 . . .	9
2.5	Создание исполняемого файла . . . . .	10
2.6	Текст измененной программы . . . . .	11
2.7	Создание исполняемого файла . . . . .	12
2.8	Текст программы из листинга 8.3 . . . . .	13
2.9	Создание исполняемого файла . . . . .	14
2.10	Создание файла листинга . . . . .	14
2.11	Открытие файла листинга . . . . .	14
2.12	Трансляция файла листинга с удалением операнда на 42 строке .	15
2.13	Текстовый редактор mscedit . . . . .	15
2.14	Программа для нахождения наименьшей переменной (1) . . . . .	16
2.15	Программа для нахождения наименьшей переменной (2) . . . . .	18
2.16	Создание исполняемого файла . . . . .	19
2.17	Текст программы задания 2 (1) . . . . .	20
2.18	Текст программы задания 2 (2) . . . . .	21
2.19	Создание исполняемого файла . . . . .	22

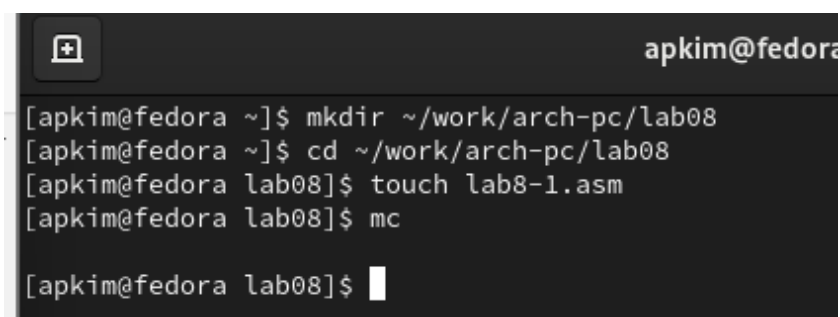
## Список таблиц

# 1 Цель работы

Цель данной работы: изучить команды условного и безусловного перехода, приобрести навыки написания программ с использованием перехода, познакомиться с назначением и структурой файла листинга.

## 2 Выполнение лабораторной работы

Создаем каталог для программ лабораторной работы №8, переходим в него и создаем файл lab8-1.asm (рис. 2.1)

A screenshot of a terminal window with a dark background. The window title bar shows a window icon and the text 'apkim@fedora'. The terminal displays the following commands and their outputs:

```
[apkim@fedora ~]$ mkdir ~/work/arch-pc/lab08
[apkim@fedora ~]$ cd ~/work/arch-pc/lab08
[apkim@fedora lab08]$ touch lab8-1.asm
[apkim@fedora lab08]$ mc
[apkim@fedora lab08]$
```

Рис. 2.1: Создание каталога и файла

Введем в файл lab8-1.asm текст программы из листинга 8.1 (рис. 2.2)

```
GNU nano 5.8
#include 'in_out.asm'
SECTION .data
msg1: DB 'Сообщение №1',0
msg2: DB 'Сообщение №2',0
msg3: DB 'Сообщение №3',0

SECTION .text
GLOBAL _start
_start:

    jmp _label2

_label1:
    mov eax, msg1
    call sprintLF

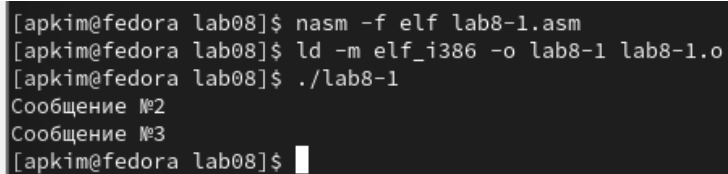
_label2:
    mov eax, msg2
    call sprintLF

_label3:
    mov eax, msg3
    call sprintLF

_end:
    call quit
```

Рис. 2.2: Текст программы из листинга 8.1

Создаем исполняемый файл и запускаем его. (рис. 2.3)



```
[apkim@fedora lab08]$ nasm -f elf lab8-1.asm
[apkim@fedora lab08]$ ld -m elf_i386 -o lab8-1 lab8-1.o
[apkim@fedora lab08]$ ./lab8-1
Сообщение №2
Сообщение №3
[apkim@fedora lab08]$
```

Рис. 2.3: Создание исполняемого файла

Далее в текст программы после вывода сообщения №2 добавим инструкцию `jmp` с меткой `_label1` и после вывода сообщения №1 добавим инструкцию `jmp` с меткой `_end`. Изменим текст программы в соответствии с листингом 8.2 (рис. 2.4)



```
GNU nano 5.8
%include 'in_out.asm'
SECTION .data
msg1: DB 'Сообщение №1',0
msg2: DB 'Сообщение №2',0
msg3: DB 'Сообщение №3',0

SECTION .text
GLOBAL _start
_start:

    jmp _label2

_label1:
    mov eax, msg1
    call sprintf
    jmp _end

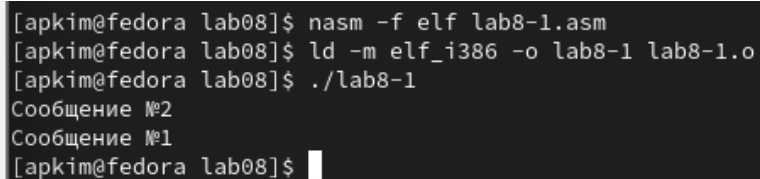
_label2:
    mov eax, msg2
    call sprintf
    jmp _label1

_label3:
    mov eax, msg3
    call sprintf

_end:
    call quit
```

Рис. 2.4: Изменяем текст программы в соответствии с листингом 8.2

Создаем исполняемый файл и проверяем его работу (рис. 2.5)

A terminal window with a dark background and light-colored text. The prompt is [apkim@fedora lab08]\$. The first command is nasm -f elf lab8-1.asm. The second command is ld -m elf\_i386 -o lab8-1 lab8-1.o. The third command is ./lab8-1. The output shows two lines of text: Сообщение №2 and Сообщение №1. The prompt returns to [apkim@fedora lab08]\$.

```
[apkim@fedora lab08]$ nasm -f elf lab8-1.asm
[apkim@fedora lab08]$ ld -m elf_i386 -o lab8-1 lab8-1.o
[apkim@fedora lab08]$ ./lab8-1
Сообщение №2
Сообщение №1
[apkim@fedora lab08]$
```

Рис. 2.5: Создание исполняемого файла

Изменим текст программы, добавив и изменив инструкцию jmp, чтобы сообщения выводились с 3-го по 1-ый (рис. 2.6)

```
GNU nano 5.8 /
#include 'in_out.asm'
SECTION .data
msg1: DB 'Сообщение №1',0
msg2: DB 'Сообщение №2',0
msg3: DB 'Сообщение №3',0

SECTION .text
GLOBAL _start
_start:

jmp _label3

_label1:
    mov eax, msg1
    call sprintfLF
    jmp _end

_label2:
    mov eax, msg2
    call sprintfLF
    jmp _label1

_label3:
    mov eax, msg3
    call sprintfLF
    jmp _label2

_end:
    call quit
```

Рис. 2.6: Текст измененной программы

Создаем исполняемый файл и проверяем его работу (рис. 2.7)

```
[apkim@fedora lab08]$ nasm -f elf lab8-1.asm
[apkim@fedora lab08]$ ld -m elf_i386 -o lab8-1 lab8-1.o
[apkim@fedora lab08]$ ./lab8-1
Сообщение №3
Сообщение №2
Сообщение №1
[apkim@fedora lab08]$ mc
[apkim@fedora lab08]$
```

Рис. 2.7: Создание исполняемого файла

Создаем файл lab8-2.asm, вводим в него текст программы из листинга 8.3 (рис. 2.8)

```

max resb 10
B    resb 10

SECTION .text
global _start
_start:

mov eax,msg1
call sprint

mov ecx,B
mov edx,10
call sread

mov eax,B
call atoi
mov [B],eax

mov ecx,[A]
mov [max],ecx

cmp ecx,[C]
jg check_B
mov ecx,[C]
mov [max],ecx

check_B:

mov eax,max
call atoi
mov [max],eax

mov ecx,[max]
cmp ecx,[B]
jg fin
mov ecx,[B]
mov [max],ecx

fin:
mov eax,msg2
call sprint
mov eax,[max]
call iprintLF
call quit

```

Рис. 2.8: Текст программы из листинга 8.3

Создаем исполняемый файл и проверяем его (рис. 2.9)

```
[apkim@fedora lab08]$ nasm -f elf lab8-2.asm
[apkim@fedora lab08]$ ld -m elf_i386 -o lab8-2 lab8-2.o
[apkim@fedora lab08]$ ./lab8-2
Введите B: 10
Наибольшее число: 50
[apkim@fedora lab08]$
```

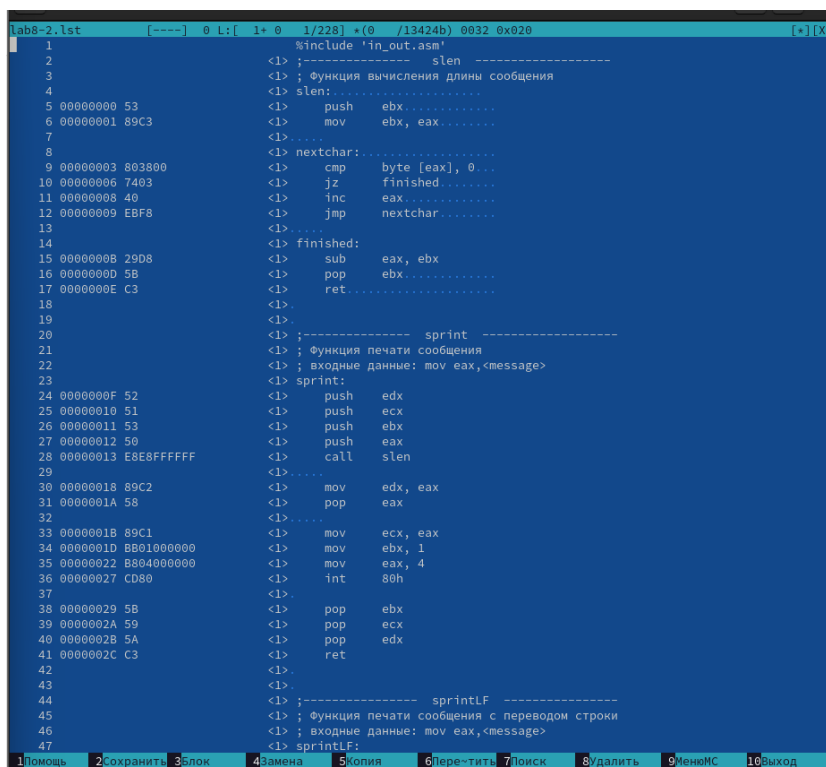
Рис. 2.9: Создание исполняемого файла

Далее создаем файл листинга для программы из файла `lab8-2.asm` (рис. 2.10)

```
[apkim@fedora lab08]$ nasm -f elf -l lab8-2.lst lab8-2.asm
[apkim@fedora lab08]$ mcedit lab8-2.lst
[apkim@fedora lab08]$
```

Рис. 2.10: Создание файла листинга

Открываем файл листинга с помощью текстового редактора `mcedit` (рис. 2.11)



```
lab8-2.lst  [----]  0 L: [ 1+ 0 1/228] * (0 /13424b) 0032 0x020 [X]
1      %include 'in_out.asm'
2      <I> ;----- slen -----
3      <I> ; Функция вычисления длины сообщения
4      <I> slen:
5      00000000 53      <I>      push     ebx
6      00000001 89C3    <I>      mov     ebx, eax
7      <I>
8      <I> nextchar:
9      00000003 803800    <I>      cmp     byte [eax], 0
10     00000006 7403    <I>      jz      finished
11     00000008 40      <I>      inc     eax
12     00000009 EBF8    <I>      jmp     nextchar
13     <I>
14     <I> finished:
15     0000000B 29D8    <I>      sub     eax, ebx
16     0000000D 5B      <I>      pop     ebx
17     0000000E C3      <I>      ret
18     <I>
19     <I>
20     <I> ;----- sprint -----
21     <I> ; Функция печати сообщения
22     <I> ; входные данные: mov eax, <message>
23     <I> sprint:
24     0000000F 52      <I>      push     edx
25     00000010 51      <I>      push     ecx
26     00000011 53      <I>      push     ebx
27     00000012 50      <I>      push     eax
28     00000013 E8E8FFFFFF <I>      call    slen
29     <I>
30     00000018 89C2    <I>      mov     edx, eax
31     0000001A 58      <I>      pop     eax
32     <I>
33     0000001B 89C1    <I>      mov     ecx, eax
34     0000001D B801000000 <I>      mov     ebx, 1
35     00000022 B804000000 <I>      mov     eax, 4
36     00000027 CD80    <I>      int     80h
37     <I>
38     00000029 5B      <I>      pop     ebx
39     0000002A 59      <I>      pop     ecx
40     0000002B 5A      <I>      pop     edx
41     0000002C C3      <I>      ret
42     <I>
43     <I>
44     <I> ;----- sprintLF -----
45     <I> ; Функция печати сообщения с переводом строки
46     <I> ; входные данные: mov eax, <message>
47     <I> sprintLF:
1Помощь 2Сохранить 3Слук 4Замена 5Копия 6Пере-тить 7Поиск 8Удалить 9МенюМС 10Выход
```

Рис. 2.11: Открытие файла листинга

Открываем файл с программой lab8-2.asm и на строке 39 удаляем операнд. Выполняем трансляции с получением файла листинга (рис. 2.12)

```
[apkim@fedora lab08]$ nasm -f elf -l lab8-2.lst lab8-2.asm
lab8-2.asm:42: error: invalid combination of opcode and operands
[apkim@fedora lab08]$ mc
[apkim@fedora lab08]$
```

Рис. 2.12: Трансляция файла листинга с удалением операнда на 42 строке

Открываем текстовый редактор mcedit (рис. 2.13)

```
37 00000130 B8[00000000]      mov eax,max
38 00000135 E862FFFFFF      call atoi
39 0000013A A3[00000000]      mov [max],eax
40 .....
41 0000013F 8B0D[00000000]      mov ecx,[max]
42                          cmp ecx
43                          error: invalid combination of opcode and operands
44 00000145 7F0C      jg fin
45 00000147 8B0D[0A000000]      mov ecx,[B]
46 0000014D 890D[00000000]      mov [max],ecx
47 .....      fin:
```

Рис. 2.13: Текстовый редактор mcedit

Задание для самостоятельной работы. 1 задание. Следует написать программу, которая из трех целочисленных переменных находит наименьшее. (рис. 2.14)

```

GNU nano 5.8
#include 'in_out.asm'
SECTION .data
A dd '95'
B dd '2'
C dd '61'
msg2 db "Наименьшее число: ",0h

SECTION .bss
min resb 10

SECTION .text
global _start
_start:

mov eax,B
call atoi
mov [B],eax

mov ecx,[A]
mov [min],ecx

cmp ecx,[C]
jl check_B
mov ecx,[C]
mov [min],ecx

check_B:

mov eax,min
call atoi
mov [min],eax

mov ecx,[min]
cmp ecx,[B]
jl fin
mov ecx,[B]
mov [min],ecx

fin:
mov eax,msg2
call sprint
mov eax,[min]
call atoi
call iprintLF

```

Рис. 2.14: Программа для нахождения наименьшей переменной (1)



Текст программы задания №1 (рис. 2.15)

```

GNU nano 5.8
A dd '95'
B dd '2'
C dd '61'
msg2 db "Наименьшее число: ",0h

SECTION .bss
min resb 10

SECTION .text
global _start
_start:

mov eax,B
call atoi
mov [B],eax

mov ecx,[A]
mov [min],ecx

cmp ecx,[C]
jl check_B
mov ecx,[C]
mov [min],ecx

check_B:

mov eax,min
call atoi
mov [min],eax

mov ecx,[min]
cmp ecx,[B]
jl fin
mov ecx,[B]
mov [min],ecx

fin:
mov eax,msg2
call sprint
mov eax,[min]
call atoi
call iprintLF
call quit

```

Рис. 2.15: Программа для нахождения наименьшей переменной (2)

Создаем исполняемый файл и проверяем его. У меня был вариант-20, значит вывести программа должна была цифру 2, но она вывела 50, что в таблице ASCII соответствует символу “2” (рис. 2.16)

```
[apkim@fedora lab08]$ nasm -f elf -l lab8-3.lst lab8-3.asm
[apkim@fedora lab08]$ ld -m elf_i386 -o lab8-3 lab8-3.o
[apkim@fedora lab08]$ ./lab8-3
Наименьшее число: 50
```

Рис. 2.16: Создание исполняемого файла

2 задание. Необходимо написать программу, которая для введенных с клавиатуры значений  $x$  и  $a$  вычисляет значение заданной функции  $f(x)$  и выводит результат вычислений. (рис. 2.17)

```

GNU nano 5.8
%include 'in_out.asm'
SECTION .data
msg1 db "Введите x: ",0h
msg2 db "Введите a: ",0h

SECTION .bss
x resb 10
a resb 10

SECTION .text
global _start
_start:

mov eax, msg1
call sprint

mov ecx, x
mov edx, 10
call sread

mov eax, x
call atoi
mov [x], eax

mov eax, msg2
call sprint

mov ecx, a
mov edx, 10
call sread

mov eax, a
call atoi
mov [a], eax

mov ecx, [x]
cmp ecx, [a]
jae net
mov eax, 5
call iprintLF
jmp end

net:
mov eax, [x]

```

Рис. 2.17: Текст программы задания 2 (1)

Продолжение программы (рис. 2.18)

```
net:  
mov  eax,[x]  
mov  ebx,[a]  
sub  eax,ebx  
call iprintLF  
jmp  end  
  
end:  
call quit
```

Рис. 2.18: Текст программы задания 2 (2)

Создаем исполняемый файл и проверяем его (рис. 2.19)

```
[apkim@fedora lab08]$ nasm -f elf lab8-4.asm
[apkim@fedora lab08]$ ld -m elf_i386 -o lab8-4 lab8-2.o
ld: невозможно найти lab8-2.o: Нет такого файла или каталога
[apkim@fedora lab08]$ ld -m elf_i386 -o lab8-4 lab8-4.o
[apkim@fedora lab08]$ ./lab8-4
Введите x: 1
Введите a: 2
5
[apkim@fedora lab08]$ ./lab8-4
Введите x: 2
Введите a: 1
1
[apkim@fedora lab08]$ ./lab8-4
Введите x: 10
Введите a: 10
0
[apkim@fedora lab08]$ ./lab8-4
Введите x: 7
Введите a: 100
5
```

Рис. 2.19: Создание исполняемого файла

## 3 Выводы

В ходе данной лабораторной работы я изучила команды условного и безусловного перехода, приобрела навыки написания программ с использованием перехода, познакомилась с назначением и структурой файла листинга.

## **Список литературы**