

Отчёт по лабораторной работе 8

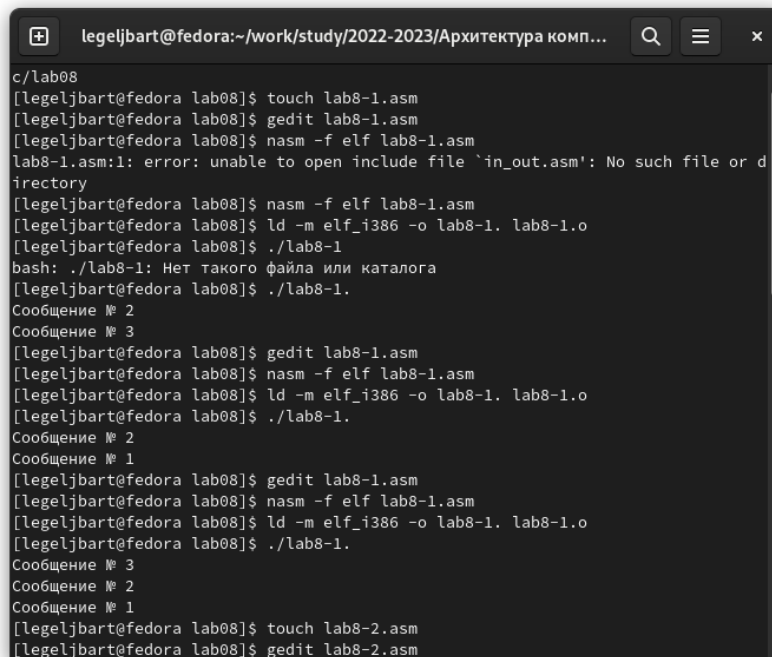
Безусловный и условный переход в NASM

Лев Евгеньевич Гельбарт

1 Цель работы

Изучение команд условного и безусловного переходов. Приобретение навыков написания программ с использованием переходов. Знакомство с назначением и структурой файла листинга.

2 Выполнение лабораторной работы



```
legeljbart@fedora:~/work/study/2022-2023/Архитектура комп...
c/lab08
[legeljbart@fedora lab08]$ touch lab8-1.asm
[legeljbart@fedora lab08]$ gedit lab8-1.asm
[legeljbart@fedora lab08]$ nasm -f elf lab8-1.asm
lab8-1.asm:1: error: unable to open include file `in_out.asm': No such file or d
irectory
[legeljbart@fedora lab08]$ nasm -f elf lab8-1.asm
[legeljbart@fedora lab08]$ ld -m elf_i386 -o lab8-1. lab8-1.o
[legeljbart@fedora lab08]$ ./lab8-1
bash: ./lab8-1: Нет такого файла или каталога
[legeljbart@fedora lab08]$ ./lab8-1.
Сообщение № 2
Сообщение № 3
[legeljbart@fedora lab08]$ gedit lab8-1.asm
[legeljbart@fedora lab08]$ nasm -f elf lab8-1.asm
[legeljbart@fedora lab08]$ ld -m elf_i386 -o lab8-1. lab8-1.o
[legeljbart@fedora lab08]$ ./lab8-1.
Сообщение № 2
Сообщение № 1
[legeljbart@fedora lab08]$ gedit lab8-1.asm
[legeljbart@fedora lab08]$ nasm -f elf lab8-1.asm
[legeljbart@fedora lab08]$ ld -m elf_i386 -o lab8-1. lab8-1.o
[legeljbart@fedora lab08]$ ./lab8-1.
Сообщение № 3
Сообщение № 2
Сообщение № 1
[legeljbart@fedora lab08]$ touch lab8-2.asm
[legeljbart@fedora lab08]$ gedit lab8-2.asm
```

По образцу пишем программы, выдающие сообщения 2 и 3, 2 и 1, а затем 3, 2 и 1 (рис. [001]).

```

[legeljbart@fedora lab08]$ gedit lab8-2.asm
[legeljbart@fedora lab08]$ nasm -f elf lab8-2.asm
lab8-2.asm:23: error: parser: instruction expected
lab8-2.asm:33: warning: label alone on a line without a colon might be in error
[-w+label-orphan]
[legeljbart@fedora lab08]$ gedit lab8-2.asm
[legeljbart@fedora lab08]$ nasm -f elf lab8-2.asm
lab8-2.asm:23: error: parser: instruction expected
lab8-2.asm:33: warning: label alone on a line without a colon might be in error
[-w+label-orphan]
[legeljbart@fedora lab08]$ gedit lab8-2.asm
[legeljbart@fedora lab08]$ nasm -f elf lab8-2.asm
lab8-2.asm:29: error: symbol 'check' not defined
[legeljbart@fedora lab08]$ gedit lab8-2.asm
[legeljbart@fedora lab08]$ nasm -f elf lab8-2.asm
[legeljbart@fedora lab08]$ ld -m elf_i386 -o lab8-2. lab8-2.o
[legeljbart@fedora lab08]$ ./lab8-2.
Введите В: 26
Наибольшее число: 50
[legeljbart@fedora lab08]$ ./lab8-2.
Введите В: 100
Наибольшее число: 100
[legeljbart@fedora lab08]$ nasm -f elf -l lab8-2.lst lab8-2.asm
[legeljbart@fedora lab08]$ mcedit lab8-2.lst

[legeljbart@fedora lab08]$ gedit lab8-2.asm
[legeljbart@fedora lab08]$ nasm -f elf -l lab8-2.lst lab8-2.asm
lab8-2.asm:31: error: invalid combination of opcode and operands

```

По следующему шаблону пишем код, выдающий наибольшее число из 20, 50 и введенного числа (рис. [002]). Внизу видно, что получится при удалении одной из операнд

```

lab8-2.lst  [----]  0 L: [ 43+ 0 43/225] *(2559/13340b) 0032 0x020 [*][X]
43          <1> .
44          <1> ;----- sprintLF -----
45          <1> ; Функция печати сообщения с переводом с
46          <1> ; входные данные: mov eax,<message>
47          <1> sprintLF:
48 0000002D E8DDFFFFFF <1> call sprint
49          <1> .
50 00000032 50 <1> push eax
51 00000033 B80A000000 <1> mov eax, 0Ah
52 00000038 50 <1> push eax
53 00000039 89E0 <1> mov eax, esp
54 0000003B E8CFFFFFFF <1> call sprint
55 00000040 58 <1> pop eax
56 00000041 58 <1> pop eax
57 00000042 C3 <1> ret
58          <1> .
59          <1> ;----- sread -----
60          <1> ; Функция считывания сообщения
61          <1> ; входные данные: mov eax,<buffer>, mov
62          <1> sread:
63 00000043 53 <1> push ebx
64 00000044 50 <1> push eax
65          <1> ....
66 00000045 B800000000 <1> mov ebx, 0
67 0000004A B803000000 <1> mov eax, 3
68 0000004F CD80 <1> int 80h

```

в какой-либо строке.

Откроем листинг(рис. [003]). Рассмотрим три любые строки 55 - строка 00000040 - адрес 58 - машинный код `pop eax` - текст программы 57 -

строка 00000042 - адрес C3 - машинный код ret - текст программы 63 -
 строка 00000043 - адрес 53 - машинный код push ebx - текст программы

```

legeljbart@fedora:~/work/study/2022-2023/Архитектура комп...
lab8-2.lst  [----]  0 L:[195+22 217/226] *(12954/13427b) 0032 0x020[*] [X]
20.....
21 00000101 B8[0A000000]      mov eax,B
22 00000106 E891FFFFFF      call atoi
23 0000010B A3[0A000000]      mov [B],eax
24.....
25 00000110 8B0D[36000000]      mov ecx,[A]
26 00000116 890D[00000000]      mov [max],ecx
27.....
28 0000011C 3B0D[3A000000]      cmp ecx,[C]
29 00000122 7F06            jg check_B
30 00000124 8B0D[3A000000]      mov ecx,[C]
31      mov [max]
31      *****      error: invalid combination of opcode an
32.....
33      check_B:
34 0000012A B8[00000000]      mov eax,max
35 0000012F E868FFFFFF      call atoi
36 00000134 A3[00000000]      mov [max],eax
37.....
38 00000139 8B0D[00000000]      mov ecx,[max]
39 0000013F 3B0D[0A000000]      cmp ecx,[B]
40 00000145 7F0C            jg fin
41 00000147 8B0D[0A000000]      mov ecx,[B]
42 0000014D 890D[00000000]      mov [max],ecx
43.....
44      fin:

```

При удалении операнды

```

[legeljbart@fedora lab08]$ gedit l
[legeljbart@fedora lab08]$ nasm -f
[legeljbart@fedora lab08]$ ld -m e
[legeljbart@fedora lab08]$ ./lab8-
Введите B: 62
Наименьшее число: 54

```

в конце листинга добавляется сообщение об ошибке (рис. [005]).

(Вариант5) С помощью нетрудный манипуляций в приложенном коде получим

```
lab8-3.asm
~/work/study/2022-2023/Архитектура компьютера/arc...

Открыть ▾ +

20
21 mov eax,B
22 call atoi
23 mov [B],eax
24
25 mov ecx,[A]
26 mov [min],ecx
27
28 cmp ecx,[C]
29 jg check_B
30 mov ecx,[A]
31 mov [min],ecx
32
33 check_B:
34 mov eax,min
35 call atoi
36 mov [min],eax
37
38 mov ecx,[min]
39 cmp [B],ecx
40 jg fin
41 mov ecx,[B]
42 mov [min],ecx
43
44 fin:
45 mov eax,msg2
46 call sprint
47 mov eax,[min]
48 call iprintLF
49 call quit

Matlab ▾ Ширина табуляции: 8 ▾ C
```

программу, выдающую наименьшее число (рис. [006]).

```
lab8-4.asm
~/work/study/2022-2023/Архитектура к...

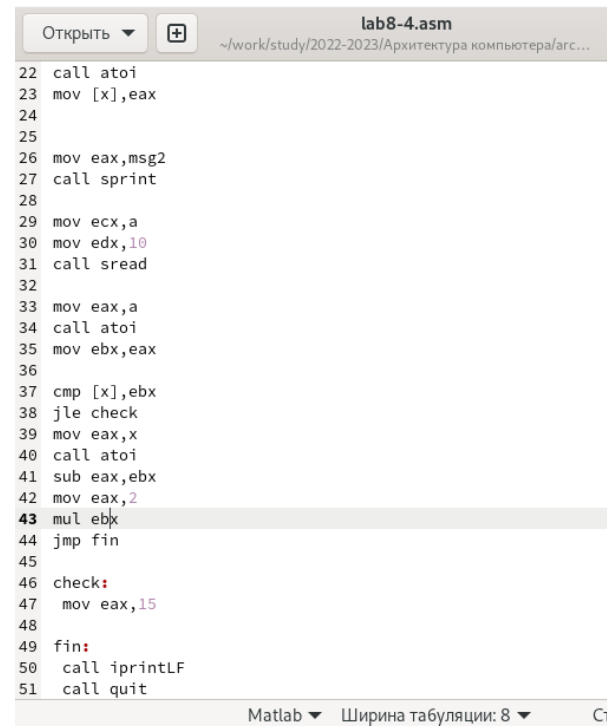
Открыть ▾ +

22 call atoi
23 mov [x],eax
24
25
26 mov eax,msg2
27 call sprint
28
29 mov ecx,a
30 mov edx,10
31 call sread
32
33 mov eax,a
34 call atoi
35 mov ebx,eax
36
37 cmp [x],ebx
38 jle check
39 mov eax,x
40 call atoi
41 sub eax,ebx
42 mov eax,2
43 mul ebx
44 jmp fin
45
46 check:
47 mov eax,15
48
49 fin:
50 call iprintLF
51 call quit

Matlab ▾ Ширина табуляц
```

Сам модифицированный код, показаны изменения (рис. [007]).

Видим, что программа, выполняющая заданную функцию, выполняется



```
lab8-4.asm
~/work/study/2022-2023/Архитектура компьютера/arc...

22 call atoi
23 mov [x],eax
24
25
26 mov eax,msg2
27 call sprintf
28
29 mov ecx,a
30 mov edx,10
31 call sread
32
33 mov eax,a
34 call atoi
35 mov ebx,eax
36
37 cmp [x],ebx
38 jle check
39 mov eax,x
40 call atoi
41 sub eax,ebx
42 mov eax,2
43 mul ebx
44 jmp fin
45
46 check:
47 mov eax,15
48
49 fin:
50 call iprintLF
51 call quit
```

правильно на двух контрольных примерах (рис. [008]).

А здесь приведен готовый текст кода (рис. [009]). # Выводы

Были изучены команды условного и безусловного переодов и были получены навыки написания программ с использованием их, а также я ознакомился с файлом листинга и его свойствами.