

Отчёт по лабораторной работе №8

Дисциплина: Архитектура компьютера

Рыжкова Ульяна Валерьевна

Содержание

Список иллюстраций

Список таблиц

1 Цель работы

Изучение команд условного и безусловного переходов, приобретение навыков написания программ с переходами, знакомство с назначением и структурой файла листинга.

2 Выполнение лабораторной работы

1. Создав каталог lab08 и файл lab8-1.asm, переписываю в файл программу из листинга 8.1. Создаю исполняемый файл и запускаю его:

```
uvr1zhkova@barmaglot:/home/ru/work/study/2022-2023/Архитектура компьютера/arch-pc/lab08> nasm -f elf lab8-1.asm
uvr1zhkova@barmaglot:/home/ru/work/study/2022-2023/Архитектура компьютера/arch-pc/lab08> ld -m elf_i386 -o lab8-1 lab8-1.o
uvr1zhkova@barmaglot:/home/ru/work/study/2022-2023/Архитектура компьютера/arch-pc/lab08> ./lab8-1
Сообщение No 2
Сообщение No 3
uvr1zhkova@barmaglot:/home/ru/work/study/2022-2023/Архитектура компьютера/arch-pc/lab08> █
```

2. Результат работы изменённого файла:

```
uvr1zhkova@barmaglot:/home/ru/work/study/2022-2023/Архитектура компьютера/arch-pc/lab08> nasm -f elf lab8-1.asm
uvr1zhkova@barmaglot:/home/ru/work/study/2022-2023/Архитектура компьютера/arch-pc/lab08> ld -m elf_i386 -o lab8-1 lab8-1.o
uvr1zhkova@barmaglot:/home/ru/work/study/2022-2023/Архитектура компьютера/arch-pc/lab08> ./lab8-1
Сообщение No 2
Сообщение No 1
```

3. Меняю программу в соответствии с требованием к результату. Программа работает корректно:

```
lab8-1.asm [----] 15 L: [ 1+25 26/ 30] *(382 / 408b) 0010 0x00A [*](X)
#include <stdio.h>
SECTION .text
msg1: DB "Сообщение No 1",0
msg2: DB "Сообщение No 2",0
msg3: DB "Сообщение No 3",0

SECTION .text
GLOBAL _start
_start:
    jmp _label3

_label1:
    mov eax, msg1
    call sprintf
    jmp _end

_label2:
    mov eax, msg2
    call sprintf
    jmp _label1

_label3:
    mov eax, msg3
    call sprintf
    jmp _label2

_end:
    call quit

uvr1zhkova@barmaglot:/home/ru/work/study/2022-2023/Архитектура компьютера/arch-pc/lab08> nasm -f elf lab8-1.asm
uvr1zhkova@barmaglot:/home/ru/work/study/2022-2023/Архитектура компьютера/arch-pc/lab08> ld -m elf_i386 -o lab8-1 lab8-1.o
uvr1zhkova@barmaglot:/home/ru/work/study/2022-2023/Архитектура компьютера/arch-pc/lab08> ./lab8-1
Сообщение No 3
Сообщение No 2
Сообщение No 1
```

5. Переписываю программу из листинга 8.3 в файл lab8-2.asm, создаю исполняемый файл и смотрю результат работы при различных значениях В:

```

uvr1hzhkova@barmaglot:/home/ru/work/study/2022-2023/Архитектура компьютера/arch-pc/lab08> nasm -f elf lab8-2.asm
uvr1hzhkova@barmaglot:/home/ru/work/study/2022-2023/Архитектура компьютера/arch-pc/lab08> ld -m elf_i386 -o lab8-2 lab8-2.o
uvr1hzhkova@barmaglot:/home/ru/work/study/2022-2023/Архитектура компьютера/arch-pc/lab08> ./lab8-2
Введите B: 10
Наибольшее число: 50
uvr1hzhkova@barmaglot:/home/ru/work/study/2022-2023/Архитектура компьютера/arch-pc/lab08> ./lab8-2
Введите B: 100
Наибольшее число: 100

```

6. Открываю полученный с помощью команды `nasm -f elf -l lab8-2.lst` `lab8-2.asm` файл листинга:

```

lab8-2.lst [----] 0 L: 1+ 0 1/226 *(0 /13446b) 0032 0x020 [*](X
1      %include 'ln_out.asm'
1      <|> ;----- slen -----
2      <|> ; Функция вычисления длины сообщения
3      <|> slen:
4 00000000 53      <|>     push    ebx
5 00000001 89C3     <|>     mov     ebx, eax
6      <|>
7      <|> nextchar:
8 00000003 803800     <|>     cmp     byte [eax], 0
9 00000006 7403     <|>     jz      finished
10 00000008 40      <|>     inc     eax
11 00000009 EBF8     <|>     jmp     nextchar
12      <|>
13      <|> finished:
14 0000000B 29D8     <|>     sub     eax, ebx
15 0000000D 5B      <|>     pop     ebx
16 0000000E C3      <|>     ret
17      <|>
18      <|>
19      <|> ;----- sprint -----
20      <|> ; Функция печати сообщения
21      <|> ; входные данные: mov eax,<message>
22      <|> sprint:
23 0000000F 52      <|>     push    edx
24 00000010 51      <|>     push    ecx
25 00000011 53      <|>     push    ebx
26 00000012 50      <|>     push    eax
27 00000013 E8E8FFFFFF <|>     call    slen
28      <|>
29 00000018 89C2     <|>     mov     edx, eax
30 0000001A 58      <|>     pop     eax
31      <|>
32 0000001B 89C1     <|>     mov     ecx, eax
33 0000001D B801000000 <|>     mov     ebx, 1
34 00000022 B804000000 <|>     mov     eax, 4
35 00000027 CD80     <|>     int     80h
36      <|>
37 00000029 5B      <|>     pop     ebx
38 0000002A 59      <|>     pop     ecx
39 0000002B 5A      <|>     pop     edx
40 0000002C C3      <|>     ret
41      <|>
42      <|>
43      <|> ;----- sprintf -----
44      <|> ; Функция печати сообщения с переводом строки
45      <|> ; входные данные: mov eax,<message>
46      <|> sprintf:

```

Рассмотрим три строки:

23 строка: 00000101 (адрес) B80A000000 (машинный код) `mov eax,B` (в регистр `eax` вносим значение `B`)

40 строка: 00000145 (адрес) 3B0D0A000000 (машинный код) `cmp ecx,B` (сравниваем значение регистра `ecx` и значение `B`)

21 строка: 000000FC (адрес) E842FFFFFF (машинный код) `call sread` (считываем сообщение)

```

19 000000F2 B9[0A000000]      mov ecx,B
20 000000F7 BA0A000000      mov edx,10
21 000000FC E842FFFFFF      call sread
22
23 00000101 B8[0A000000]      mov eax,B
24 00000106 E891FFFFFF      call atoi
25 0000010B A3[0A000000]      mov [B],eax
26
27 00000110 8B0D[36000000]      mov ecx,[A]
28 00000116 890D[00000000]      mov [max],ecx
29
30 0000011C 3B0D[3A000000]      cmp ecx,[C]
31 00000122 7F0C              jg check_B
32 00000124 8B0D[3A000000]      mov ecx,[C]
33 0000012A 890D[00000000]      mov [max],ecx
34
35      check_B:
36 00000130 B8[00000000]      mov eax,max
37 00000135 E862FFFFFF      call atoi
38 0000013A A3[00000000]      mov [max],eax
39 0000013F 8B0D[00000000]      mov ecx,[max]
40 00000145 3B0D[0A000000]      cmp ecx,[B]
41 0000014B 7F0C              jg fin
42 0000014D 8B0D[0A000000]      mov ecx,[B]
43 00000153 890D[00000000]      mov [max],ecx
44
45      fin:
46 00000159 B8[14000000]      mov eax,msg2

```

7. В тексте программы намеренно делаю ошибку:

```

lab8-2.asm [----] 11 L: [ 4+23 27/ 50] *(373 / 687b) 0010 0x00A [*][X]
msg2 db "Maximum number: ",0h
A dd 10
C dd 10

section .resb
max resb 10
B resb 10

section .text
global _start
_start:

    mov eax,msg1
    call sprintf

    mov ecx,B
    mov edx,10
    call sread

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

    mov ecx,
    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 sprintf
    mov ecx,[max]
    call sprintfLF
    call quit

```

8. Видим, что выводится ошибка:

```

uyrihzhkova@barmaglot: /home/ru/work/study/2022-2023/Архитектура компьютера/arch-pc/lab08> nasm -f elf -l lab8-2.lst lab8-2.asm
lab8-2.asm:27: error: invalid combination of opcode and operands

```


9. Листинг ошибки:

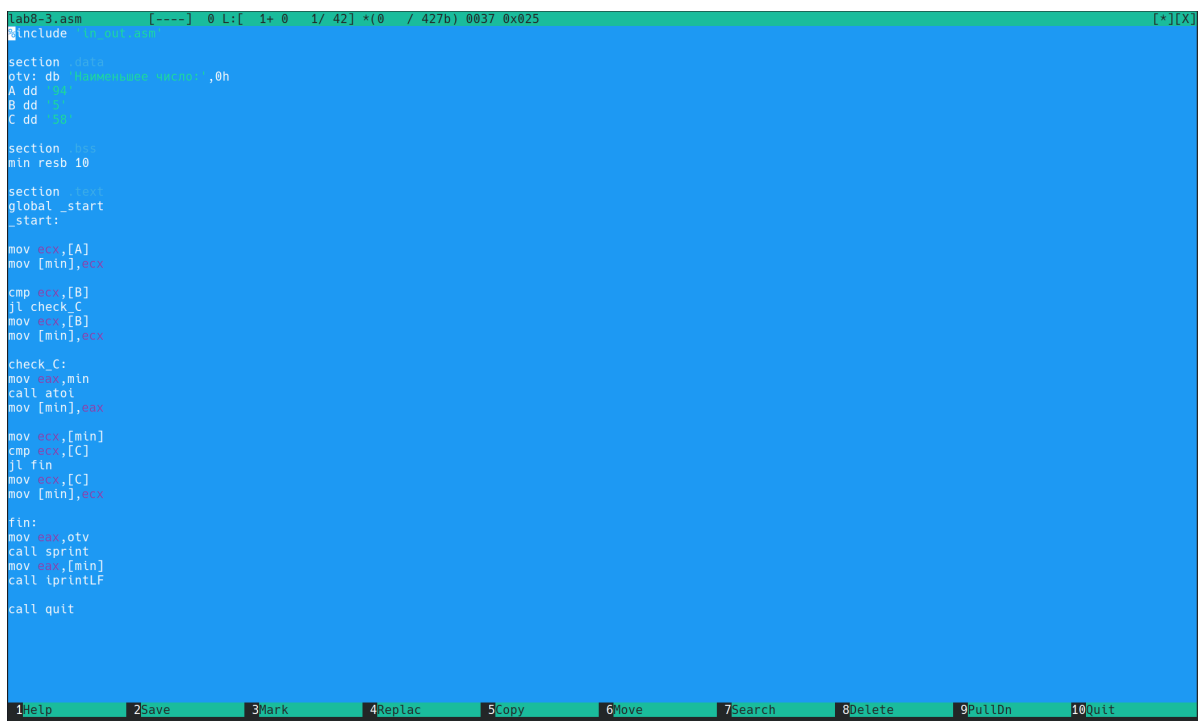
```
lab8-2.lst [B---] 90 L:[181+22 203/227] *(12298/13533b) 0010 0x00A [*][X]
6 0000003A 35300000 C dd '50'
7
8 section .bss
9 00000000 <res 0000000A> max resb 10
10 0000000A <res 0000000A> B resb 10
11
12 section .text
13 global _start
14 _start:
15
16 000000E8 B8[00000000] mov eax,msg1
17 000000ED E81DFFFFFF call sprint
18
19 000000F2 B9[0A000000] mov ecx,B
20 000000F7 BA0A000000 mov edx,10
21 000000FC E842FFFFFF call sread
22
23 00000101 B8[0A000000] mov eax,B
24 00000106 E891FFFFFF call atol
25 0000010B A3[0A000000] mov [B],eax
26
27 mov ecx
27 ***** error: invalid combination of opcode and operand:
28 00000110 890D[00000000] mov [max],ecx
29
30 00000116 3B0D[3A000000] cmp ecx,[C]
31 0000011C 7F0C jg check_B
32 0000011E 8B0D[3A000000] mov ecx,[C]
33 00000124 890D[00000000] mov [max],ecx
34
35 check_B:
36 0000012A B8[00000000] mov eax,max
37 0000012F E868FFFFFF call atol
38 00000134 A3[00000000] mov [max],eax
39 00000139 8B0D[00000000] mov ecx,[max]
40 0000013F 3B0D[0A000000] cmp ecx,[B]
41 00000145 7F0C jg fin
42 00000147 8B0D[0A000000] mov ecx,[B]
43 0000014D 890D[00000000] mov [max],ecx
44
45 fin:
46 00000153 B8[14000000] mov eax,msg2
47 00000158 E8B2FFFFFF call sprint
48 0000015D A1[00000000] mov eax,[max]
49 00000162 E81FFFFFFF call printf
50 00000167 E86FFFFFFF call quit
1Help 2Save 3Mark 4Replac 5Copy 6Move 7Search 8Delete 9PullDn 10Quit
```

10. При попытке создания выходных файлов ничего не получаем из-за ошибки:

```
uvrihzhkova@barmaglot:/home/ru/work/study/2022-2023/Архитектура компьютера/arch-pc/lab08> nasm -f elf lab8-2.asm
lab8-2.asm:27: error: invalid combination of opcode and operands
uvrihzhkova@barmaglot:/home/ru/work/study/2022-2023/Архитектура компьютера/arch-pc/lab08> ld -m elf_i386 -o lab8-2 lab8-2.o
ld: cannot find lab8-2.o: No such file or directory
uvrihzhkova@barmaglot:/home/ru/work/study/2022-2023/Архитектура компьютера/arch-pc/lab08> ./lab8-2
bash: ./lab8-2: No such file or directory
```

3 Выполнение самостоятельной работы

1. По номеру студенческого мой вариант - 3. Ищу наименьшее из набора чисел 94, 5, 58 используя программу:



```
lab8-3.asm [----] 0 L: [ 1+ 0 1/ 42] *(0 / 427b) 0037 0x025 [*][X]
#include <stdio.h>

section .data
    otv: db "Наименьшее число: ",0h
    A dd 94
    B dd 5
    C dd 58

section .bss
    min resb 10

section .text
    global _start
_start:
    mov ecx,[A]
    mov [min],ecx

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

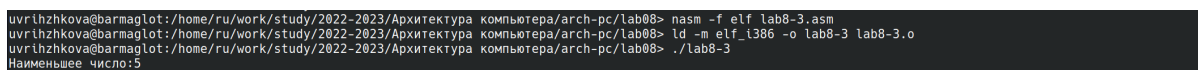
check_C:
    mov ecx,min
    call atoi
    mov [min],eax

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

fin:
    mov eax,otv
    call sprint
    mov ecx,[min]
    call lprintf
    call quit

1help 2Save 3Mark 4Replac 5Copy 6love 7Search 8Delete 9PullDn 10Quit
```

2. Программа работает корректно:



```
uvrithzkova@barmaglot: /home/ru/work/study/2022-2023/Архитектура компьютера/arch-pc/lab08> nasm -f elf lab8-3.asm
uvrithzkova@barmaglot: /home/ru/work/study/2022-2023/Архитектура компьютера/arch-pc/lab08> ld -m elf_i386 -o lab8-3 lab8-3.o
uvrithzkova@barmaglot: /home/ru/work/study/2022-2023/Архитектура компьютера/arch-pc/lab08> ./lab8-3
Наименьшее число:5
```

3. Выполнение второго задания:

```
lab8-4.asm [----] 0 L: [ 1+ 0 1/ 63] *(0 / 586b) 0037 0x025 [*][X]
#include "stdio.h"

section .data
x1: db "Введите x: ",0h
a1: db "Введите a: ",0h
otv: db "Ответ: ",0h

section .bss
x: resb 10
a: resb 10

section .text
global _start
_start:

mov eax,x1
call sprint

mov ecx,x
mov edi,10
call sread

mov eax,x
call atoi

mov [x],eax

mov eax,a1
call sprint

mov ecx,a
mov edi,10
call sread

mov eax,a
call atoi

mov [a],eax

mov ecx,3
cmp ecx,[x]
jne check_a
mov eax,[x]
mov ecx,3
mul ecx
mov edi,eax
jmp fin

1help 2save 3mark 4Replac 5copy 6move 7search 8delete 9pullDn 10quit

uvr1hzhkova@barmaglot:/home/ru/work/study/2022-2023/Архитектура компьютера/arch-pc/lab08> nasm -f elf lab8-4.asm
uvr1hzhkova@barmaglot:/home/ru/work/study/2022-2023/Архитектура компьютера/arch-pc/lab08> ld -m elf_i386 -o lab8-4 lab8-4.o
uvr1hzhkova@barmaglot:/home/ru/work/study/2022-2023/Архитектура компьютера/arch-pc/lab08> ./lab8-4
Введите x:3
Ответ:9
uvr1hzhkova@barmaglot:/home/ru/work/study/2022-2023/Архитектура компьютера/arch-pc/lab08> ./lab8-4
Введите x:1
Введите a:4
Ответ:5
```

4 Выводы

Я ознакомилась с командами переходов, научилась писать программы с ними и изучила структуру файла листинга.