

Отчёт по лабораторной работе 7

дисциплина: Архитектура компьютера

Тяпкова Альбина НММбд-04-24

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
2.1	Изучение структуры файла листинга	9
2.2	Задание для самостоятельной работы	11
3	Выводы	15

Список иллюстраций

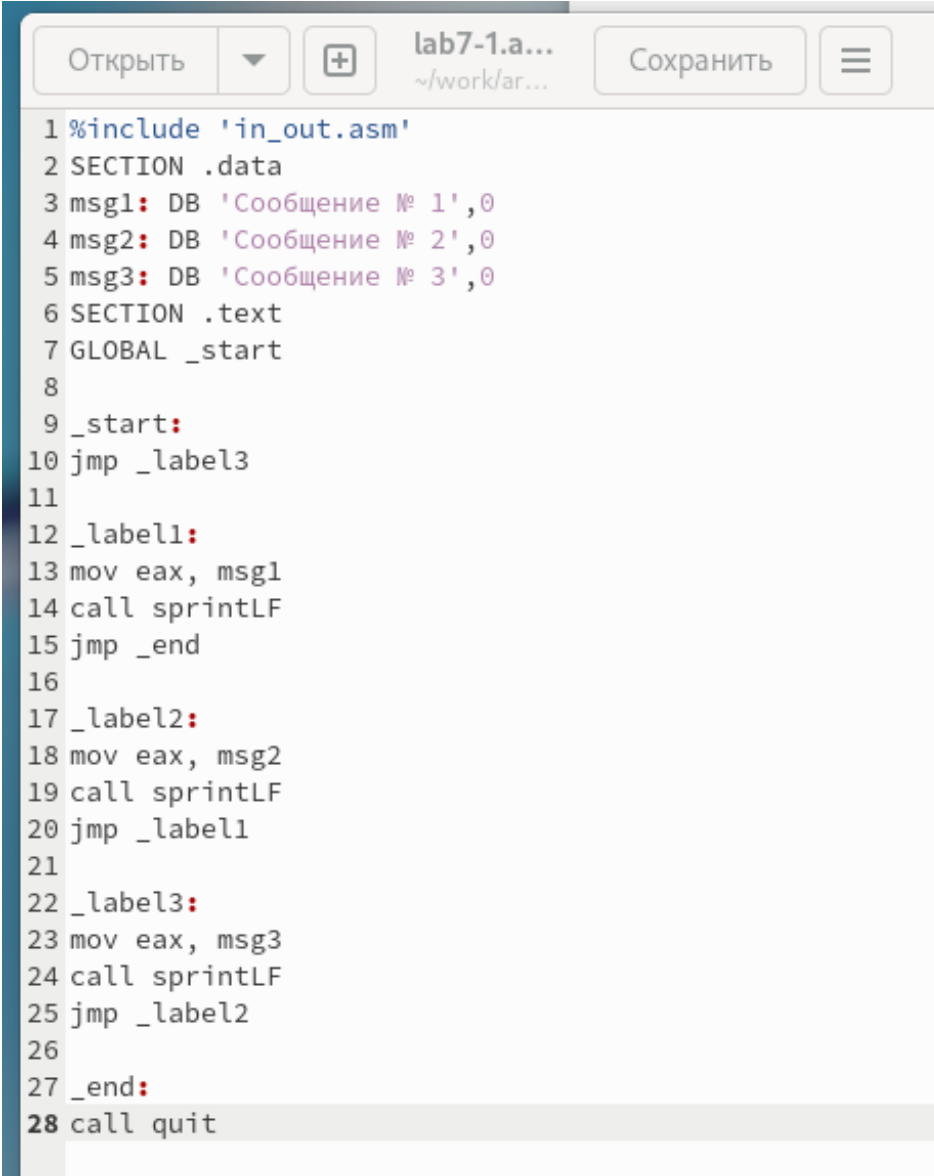
2.1	Финальная версия программы lab7-1.asm	6
2.2	Запуск программы с новой последовательностью	7
2.3	Программа для нахождения максимума в lab7-2.asm	8
2.4	Запуск программы для нахождения максимума	9
2.5	Файл листинга lab7-2	9
2.6	Ошибка трансляции lab7-2	10
2.7	Листинг с ошибкой lab7-2	11
2.8	Программа в task7-1.asm	12
2.9	Запуск task7-1.asm	12
2.10	Программа для $f(x)$ в task7-2.asm	13
2.11	Запуск task7-2.asm	14

Список таблиц

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



```
1 %include 'in_out.asm'
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
9 _start:
10 jmp _label3
11
12 _label1:
13 mov eax, msg1
14 call sprintLF
15 jmp _end
16
17 _label2:
18 mov eax, msg2
19 call sprintLF
20 jmp _label1
21
22 _label3:
23 mov eax, msg3
24 call sprintLF
25 jmp _label2
26
27 _end:
28 call quit
```

Рис. 2.1: Финальная версия программы lab7-1.asm

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

Рис. 2.2: Запуск программы с новой последовательностью

Инструкция `jmp` всегда выполняет переход, но в программах часто нужно делать условные переходы, когда действие зависит от выполнения условия. Рассмотрим пример с определением наибольшего числа из трех: А, В и С. Значения А и С заданы в программе, В вводится с клавиатуры.

Скомпилировала исполняемый файл и протестировала для разных значений В.

```
19 call sread
20 ; ----- Преобразование 'B' из символа в число
21 mov eax,B
22 call atoi
23 mov [B],eax
24 ; ----- Записываем 'A' в переменную 'max'
25 mov ecx,[A]
26 mov [max],ecx
27 ; ----- Сравниваем 'A' и 'C' (как символы)
28 cmp ecx,[C]
29 jg check_B
30 mov ecx,[C]
31 mov [max],ecx
32 ; ----- Преобразование 'max(A,C)' из символа в число
33 check_B:
34 mov eax,max
35 call atoi
36 mov [max],eax
37 ; ----- Сравниваем 'max(A,C)' и 'B' (как числа)
38 mov ecx,[max]
39 cmp ecx,[B]
40 jg fin
41 mov ecx,[B]
42 mov [max],ecx
43 ; ----- Вывод результата
44 fin:
45 mov eax, msg2
46 call sprint
47 mov eax,[max]
48 call iprintLF
49 call quit
```

Рис. 2.3: Программа для нахождения максимума в lab7-2.asm


```
artyarkova@artyarkova:~/work/arch-pc/lab07$  
artyarkova@artyarkova:~/work/arch-pc/lab07$  
artyarkova@artyarkova:~/work/arch-pc/lab07$ nasm -f elf lab7-2.asm  
artyarkova@artyarkova:~/work/arch-pc/lab07$ ld -m elf_i386 lab7-2.o -o lab7-2  
artyarkova@artyarkova:~/work/arch-pc/lab07$ ./lab7-2  
Введите B: 6  
Наибольшее число: 50  
artyarkova@artyarkova:~/work/arch-pc/lab07$ ./lab7-2  
Введите B: 60  
Наибольшее число: 60  
artyarkova@artyarkova:~/work/arch-pc/lab07$
```

Рис. 2.4: Запуск программы для нахождения максимума

2.1 Изучение структуры файла листинга

NASM обычно создает только объектный файл, но с ключом `-l` можно получить файл листинга.

Создала файл листинга для программы из `lab7-2.asm`.

```
lab7-2.lst  
~/.work/arch-pc/lab07  
lab7-2.asm x lab7-2.lst x  
174 3 00000012 00  
175 3 00000012 00  
176 4 00000013 D09DD0B0D0B8D0B1D0- msg2 db "Наибольшее число: ",0h  
177 4 0000001C BED0BBD18CD188D0B5-  
178 4 00000025 D0B520D187D0B8D181-  
179 4 0000002E D0BBD0BE3A2000  
180 5 00000035 32300000 A dd '20'  
181 6 00000039 35300000 C dd '50'  
182 7 section .bss  
183 8 00000000 <res Ah> max resb 10  
184 9 0000000A <res Ah> B resb 10  
185 10 section .text  
186 11 global _start  
187 12 _start:  
188 13 ; ----- Вывод сообщения 'Введите B: '  
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  
194 19 000000FC E842FFFFFF call sread  
195 20 ; ----- Преобразование 'B' из символа в число  
196 21 00000101 B8[0A000000] mov eax,B  
197 22 00000106 E891FFFFFF call atoi  
198 23 0000010B A3[0A000000] mov [B],eax  
199 24 ; ----- Записываем 'A' в переменную 'max'  
200 25 00000110 8B0D[35000000] mov ecx,[A]  
201 26 00000116 890D[00000000] mov [max],ecx  
202 27 ; ----- Сравниваем 'A' и 'C' (как символы)
```

Рис. 2.5: Файл листинга lab7-2

Изучила содержимое файла листинга. Приведу пример трех строк:

Строка 203

- 28 - номер строки
- 0000011C - адрес
- 3B0D[39000000] - машинный код
- `cmp ecx,[C]` - команда сравнения регистров `ecx` и переменной `C`

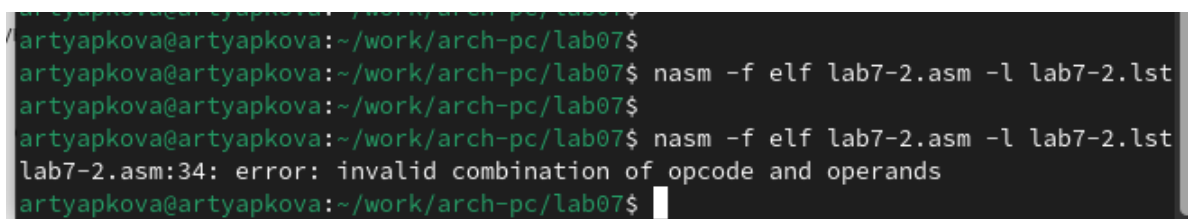
Строка 204

- 29 - номер строки
- 00000122 - адрес
- 7F0C - машинный код
- `jb check_B` - условный переход к метке `check_B`, если `>`.

Строка 205

- 30 - номер строки
- 00000124 - адрес
- 8B0D[39000000] - машинный код
- `mov ecx,[C]` - копирует значение `C` в `ecx`

Открыла `lab7-2.asm`, удалила операнд в инструкции с двумя операндами, выполнила трансляцию и получила файл листинга с ошибкой.



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

Рис. 2.6: Ошибка трансляции `lab7-2`

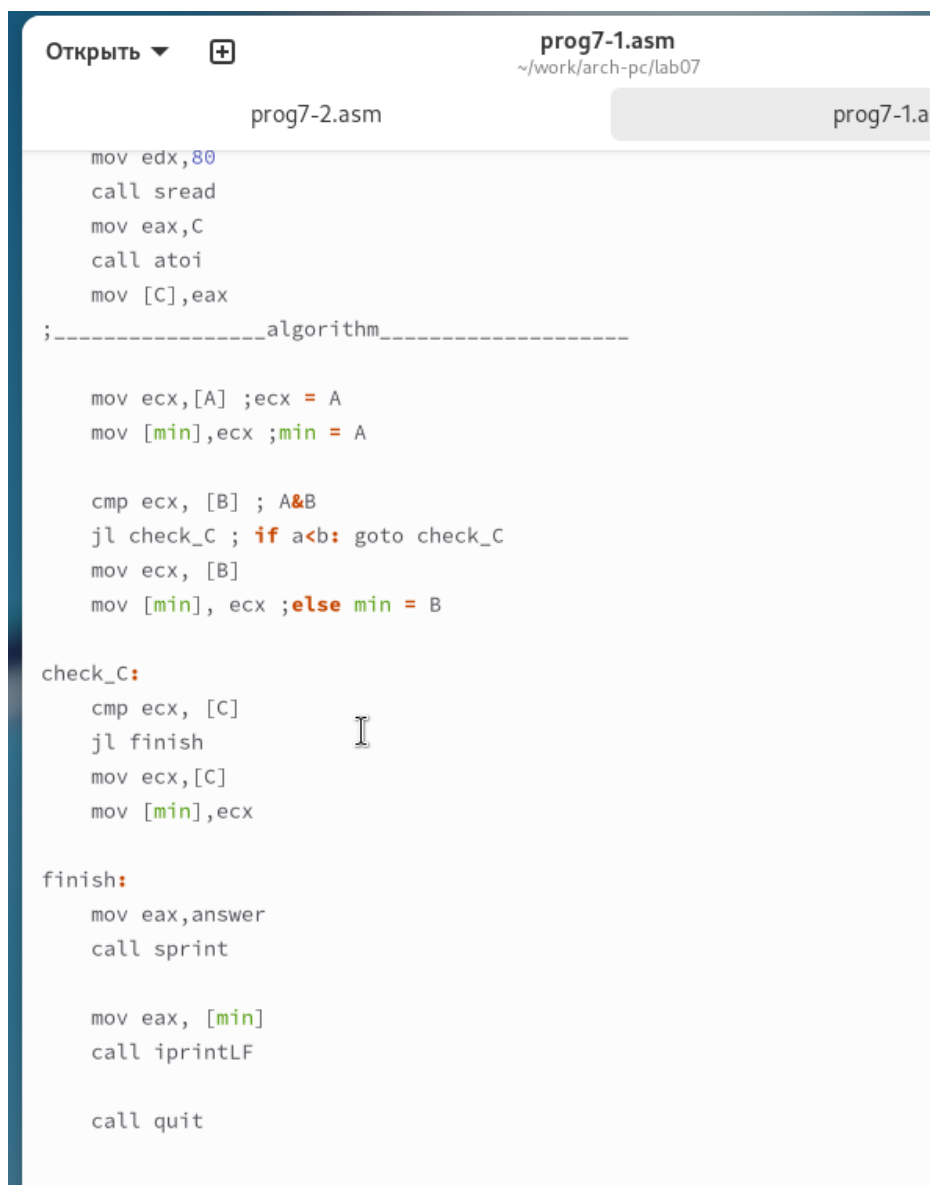
```
195 20 ; ----- Преобразование 'B' из символа в число
196 21 00000101 B8[0A000000] mov eax,B
197 22 00000106 E891FFFFFF call atoi
198 23 0000010B A3[0A000000] mov [B],eax
199 24 ; ----- Записываем 'A' в переменную 'max'
200 25 00000110 8B0D[35000000] mov ecx,[A]
201 26 00000116 890D[00000000] mov [max],ecx
202 27 ; ----- Сравниваем 'A' и 'C' (как символы)
203 28 0000011C 3B0D[39000000] cmp ecx,[C]
204 29 00000122 7F0C jg check_B
205 30 00000124 8B0D[39000000] mov ecx,[C]
206 31 0000012A 890D[00000000] mov [max],ecx
207 32 ; ----- Преобразование 'max(A,C)' из символа в число
208 33 check_B:
209 34 mov eax,
210 34 ***** error: invalid combination of opcode and operands
211 35 00000130 E867FFFFFF call atoi
212 36 00000135 A3[00000000] mov [max],eax
213 37 ; ----- Сравниваем 'max(A,C)' и 'B' (как числа)
214 38 0000013A 8B0D[00000000] mov ecx,[max]
215 39 00000140 3B0D[0A000000] cmp ecx,[B]
216 40 00000146 7F0C jg fin
217 41 00000148 8B0D[0A000000] mov ecx,[B]
218 42 0000014E 890D[00000000] mov [max],ecx
219 43 ; ----- Вывод результата
220 44 fin:
221 45 00000154 B8[13000000] mov eax, msg2
222 46 00000159 E8B1FEFFFF call sprint
223 47 0000015E A1[00000000] mov eax,[max]
224 48 00000163 E81EFFFFFF call iprintLF
225 49 00000168 E86EFFFFFF call quit
```

Рис. 2.7: Листинг с ошибкой lab7-2

Объектный файл не создавался, но в листинге видно место ошибки.

2.2 Задание для самостоятельной работы

Задание 1: Написать программу нахождения наименьшей из трех целых чисел a , b и c . Значения выбрать из таблицы 7.5 по варианту 10: 41, 62, 35. Скомпилировать и проверить программу.



```
Открыть ▾ + prog7-1.asm ~\work\arch-pc\lab07
prog7-2.asm prog7-1.a

mov edx,80
call sread
mov eax,C
call atoi
mov [C],eax

;-----algorithm-----

mov ecx,[A] ;ecx = A
mov [min],ecx ;min = A

cmp ecx, [B] ; A&B
j<B> check_C ; if a<b: goto check_C
mov ecx, [B]
mov [min], ecx ;else min = B

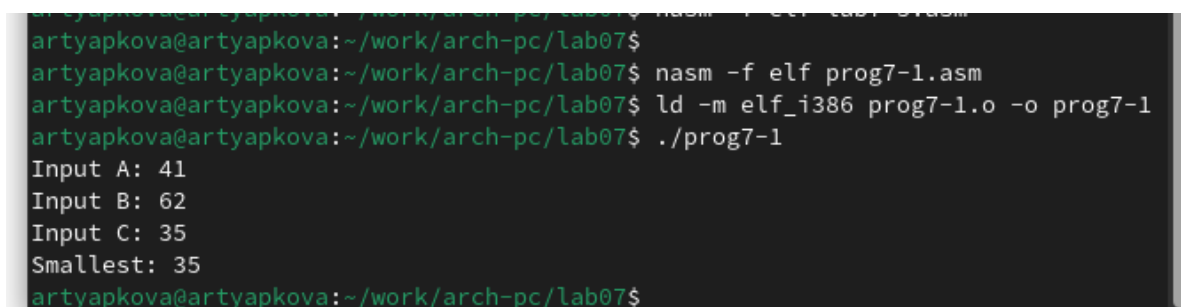
check_C:
  cmp ecx, [C]
  j<C> finish
  mov ecx,[C]
  mov [min],ecx

finish:
  mov eax,answer
  call sprint

  mov eax, [min]
  call iprintLF

  call quit
```

Рис. 2.8: Программа в task7-1.asm



```
artyapkova@artyapkova: ~\work\arch-pc\lab07$
artyapkova@artyapkova: ~\work\arch-pc\lab07$ nasm -f elf prog7-1.asm
artyapkova@artyapkova: ~\work\arch-pc\lab07$ ld -m elf_i386 prog7-1.o -o prog7-1
artyapkova@artyapkova: ~\work\arch-pc\lab07$ ./prog7-1
Input A: 41
Input B: 62
Input C: 35
Smallest: 35
artyapkova@artyapkova: ~\work\arch-pc\lab07$
```

Рис. 2.9: Запуск task7-1.asm

Задание 2: Написать программу, которая вычисляет значение функции $f(x)$ для введенных с клавиатуры значений x и a . Формулу $f(x)$ выбрать из таблицы 7.6 для варианта 10:

$$\begin{cases} x - 2, x > 2 \\ 3a, x \leq 2 \end{cases}$$

Скомпилировать и протестировать для значений из таблицы.

```
prog7-2.asm
~/work/arch-pc/lab07

mov eax,msgX
call sprint
mov ecx,X
mov edx,80
call sread
mov eax,X
call atoi
mov [X],eax

;-----algorithm-----

mov ebx, [X]
mov edx, 2
cmp ebx, edx
ja first
jmp second

first:
mov eax,[X]
sub eax,2
call iprintLF
call quit

second:
mov eax,[A]
mov ebx,3
mul ebx
call iprintLF
call quit
```

Рис. 2.10: Программа для $f(x)$ в task7-2.asm

```
artyapkova@artyapkova:~/work/arch-pc/lab07$  
artyapkova@artyapkova:~/work/arch-pc/lab07$ nasm -f elf prog7-2.asm  
artyapkova@artyapkova:~/work/arch-pc/lab07$ ld -m elf_i386 prog7-2.o -o prog7-2  
artyapkova@artyapkova:~/work/arch-pc/lab07$ ./prog7-2  
Input A: 0  
Input X: 3  
1  
artyapkova@artyapkova:~/work/arch-pc/lab07$ ./prog7-2  
Input A: 2  
Input X: 1  
6  
artyapkova@artyapkova:~/work/arch-pc/lab07$
```

Рис. 2.11: Запуск task7-2.asm

3 Выводы

Изучили команды условного и безусловного переходов, познакомились с фалом листинга.