Отчёт по лабораторной работе 7

Архитектура компьютера

Булут Умут

Содержание

Список иллюстраций

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

2.1 Реализация переходов в NASM

Создал каталог для программам лабораторной работы № 7 и файл lab7-1.asm

Инструкция jmp в NASM используется для реализации безусловных переходов. Рассмотрим пример программы с использованием инструкции jmp. Написал в файл lab7-1.asm текст программы из листинга 7.1.

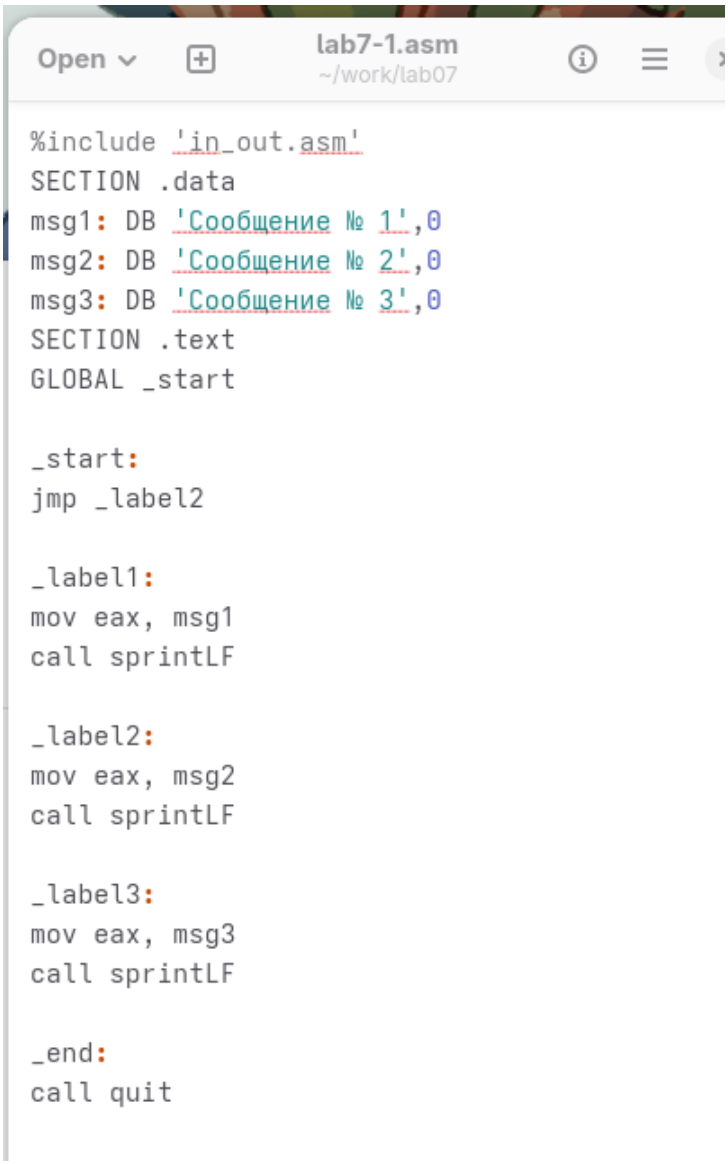


Рисунок 2.1: Программа в файле lab7-1.asm

Создал исполняемый файл и запустил его.

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

Рисунок 2.2: Запуск программы lab7-1.asm

Инструкция jmp позволяет осуществлять переходы не только вперед но и назад. Изменим программу таким образом, чтобы она выводила сначала „Сообщение № 2“, потом „Сообщение № 1“ и завершала работу. Для этого в текст программы после вывода сообщения № 2 добавим инструкцию jmp с меткой _label1 (т.е. переход к инструкциям вывода сообщения № 1) и после вывода сообщения № 1 добавим инструкцию jmp с меткой _end (т.е. переход к инструкции call quit).

Изменил текст программы в соответствии с листингом 7.2.

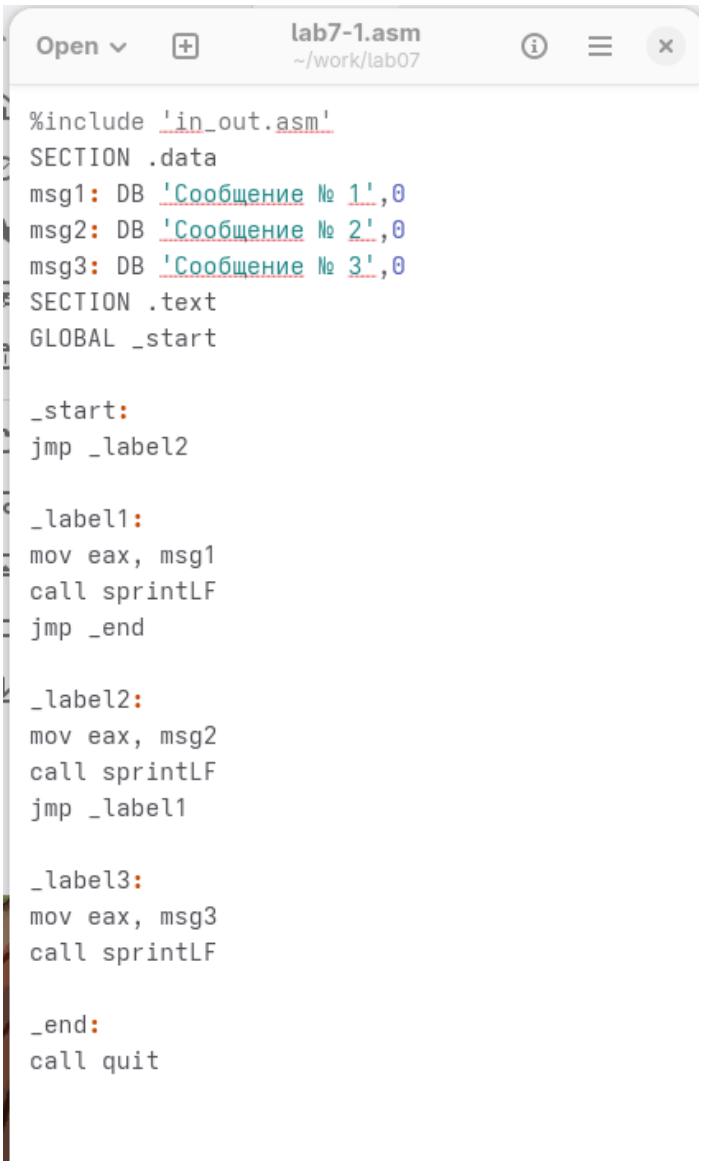


Рисунок 2.3: Программа в файле lab7-1.asm

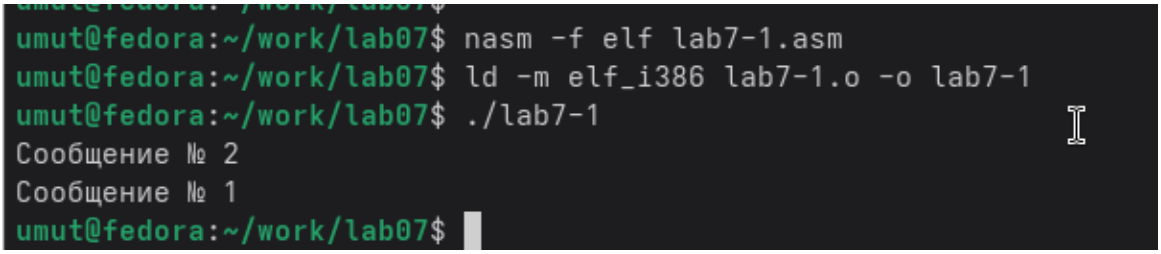


Рисунок 2.4: Запуск программы lab7-1.asm

Изменил текст программы, изменив инструкции jmp, чтобы вывод программы был следующим:

```
Сообщение № 3
Сообщение № 2
Сообщение № 1
```

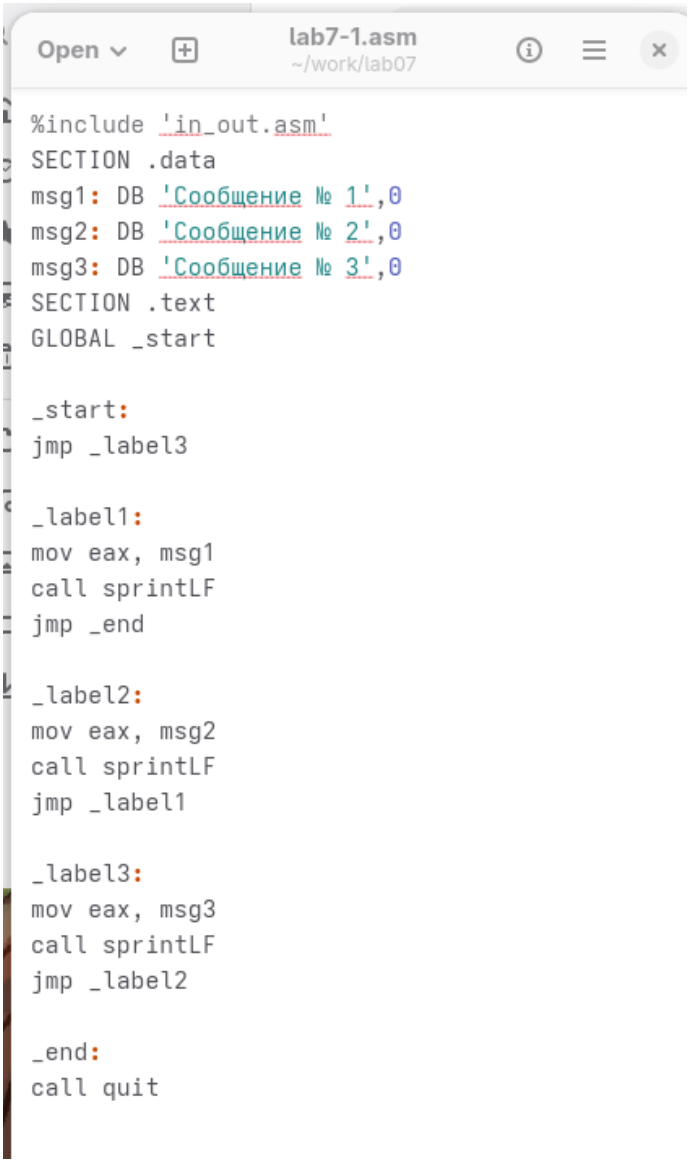


Рисунок 2.5: Программа в файле lab7-1.asm

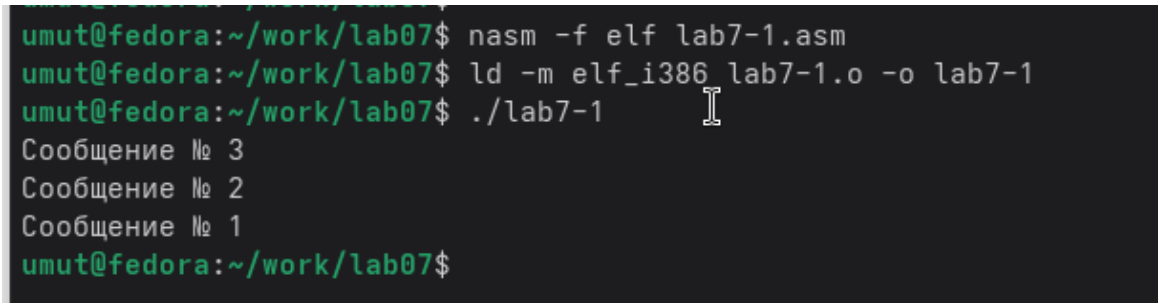


Рисунок 2.6: Запуск программы lab7-1.asm

Использование инструкции jmp приводит к переходу в любом случае. Однако, часто при написании программ необходимо использовать условные переходы, т.е. переход должен происходить если выполнено какое-либо условие. В качестве примера рассмотрим программу, которая определяет и выводит на экран наибольшую из 3 целочисленных переменных: A,B и C. Значения для A и C задаются в программе, значение B вводиться с клавиатуры.

Создал исполняемый файл и проверил его работу для разных значений B.

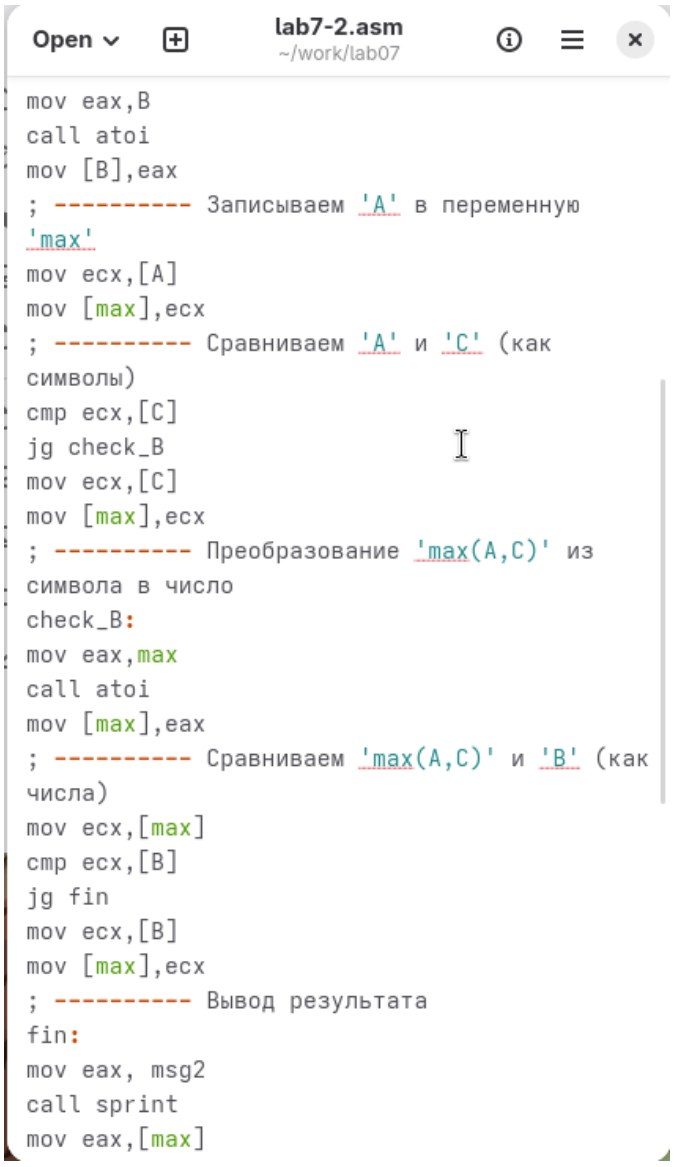


Рисунок 2.7: Программа в файле lab7-2.asm

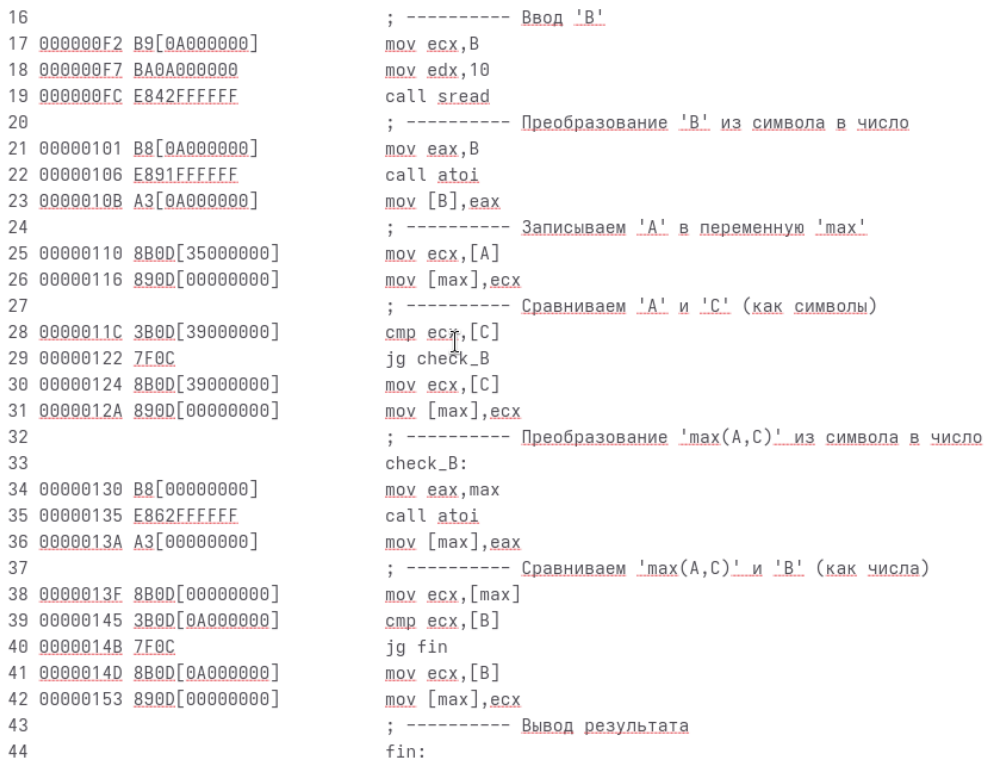
```
umut@fedora:~/work/lab07$ nasm -f elf lab7-2.asm
umut@fedora:~/work/lab07$ ld -m elf_i386 lab7-2.o -o lab7-2
umut@fedora:~/work/lab07$ ./lab7-2
Введите B: 5
Наибольшее число: 50
umut@fedora:~/work/lab07$ ./lab7-2
Введите B: 60
Наибольшее число: 60
umut@fedora:~/work/lab07$
```

Рисунок 2.8: Запуск программы lab7-2.asm

2.2 Изучение структуры файлы листинга

Обычно nasm создаёт в результате ассемблирования только объектный файл. Получить файл листинга можно, указав ключ -l и задав имя файла листинга в командной строке.

Создал файл листинга для программы из файла lab7-2.asm



```
16                                  ; ---------- Ввод 'B'
17 000000F2 B9[0A000000]            mov ecx,B
18 000000F7 BA0A000000              mov edx,10
19 000000FC E842FFFFFF              call sread
20                                  ; ---------- Преобразование 'B' из символа в число
21 00000101 B8[0A000000]            mov eax,B
22 00000106 E891FFFFFF              call atoi
23 0000010B A3[0A000000]            mov [B],eax
24                                  ; ---------- Записываем 'A' в переменную 'max'
25 00000110 8B0D[35000000]          mov ecx,[A]
26 00000116 890D[00000000]          mov [max],ecx
27                                  ; ---------- Сравниваем 'A' и 'C' (как символы)
28 0000011C 3B0D[39000000]          cmp ecx,[C]
29 00000122 7F0C                    jg check_B
30 00000124 8B0D[39000000]          mov ecx,[C]
31 0000012A 890D[00000000]          mov [max],ecx
32                                  ; ---------- Преобразование 'max(A,C)' из символа в число
33                                  check_B:
34 00000130 B8[00000000]            mov eax,max
35 00000135 E862FFFFFF              call atoi
36 0000013A A3[00000000]            mov [max],eax
37                                  ; ---------- Сравниваем 'max(A,C)' и 'B' (как числа)
38 0000013F 8B0D[00000000]          mov ecx,[max]
39 00000145 3B0D[0A000000]          cmp ecx,[B]
40 0000014B 7F0C                    jg fin
41 0000014D 8B0D[0A000000]          mov ecx,[B]
42 00000153 890D[00000000]          mov [max],ecx
43                                  ; ---------- Вывод результата
44                                  fin:
```

Рисунок 2.9: Файл листинга lab7-2

Внимательно ознакомился с его форматом и содержимым. Подробно объясню содержимое трёх строк файла листинга по выбору.

строка 203

- 28 - номер строки в подпрограмме
- 0000011C - адрес
- 3B0D[39000000] - машинный код
- cmp ecx,[C] - код программы - сравнивает регистр ecx и переменную C

строка 204

- 29 - номер строки в подпрограмме
- 00000122 - адрес

- 7F0C - машинный код
- jg check_B - код программы - если >, то переход к метке check_B

строка 205

- 30 - номер строки в подпрограмме
- 00000124 - адрес
- 8B0D[39000000] - машинный код
- mov ecx,[C] - код программы - перекладывает в регистр ecx значение переменной C

Открыл файл с программой lab7-2.asm и в инструкции с двумя операндами удалил один операнд. Выполнил трансляцию с получением файла листинга.

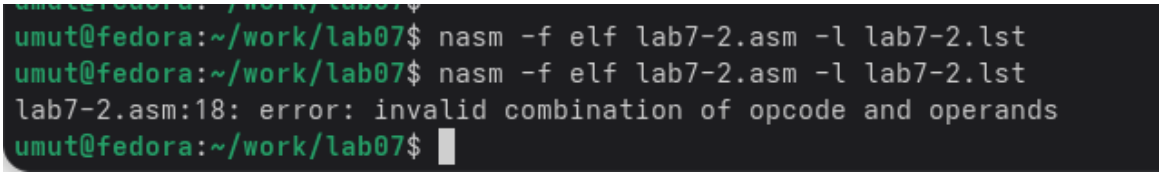


Рисунок 2.10: Ошибка трансляции lab7-2

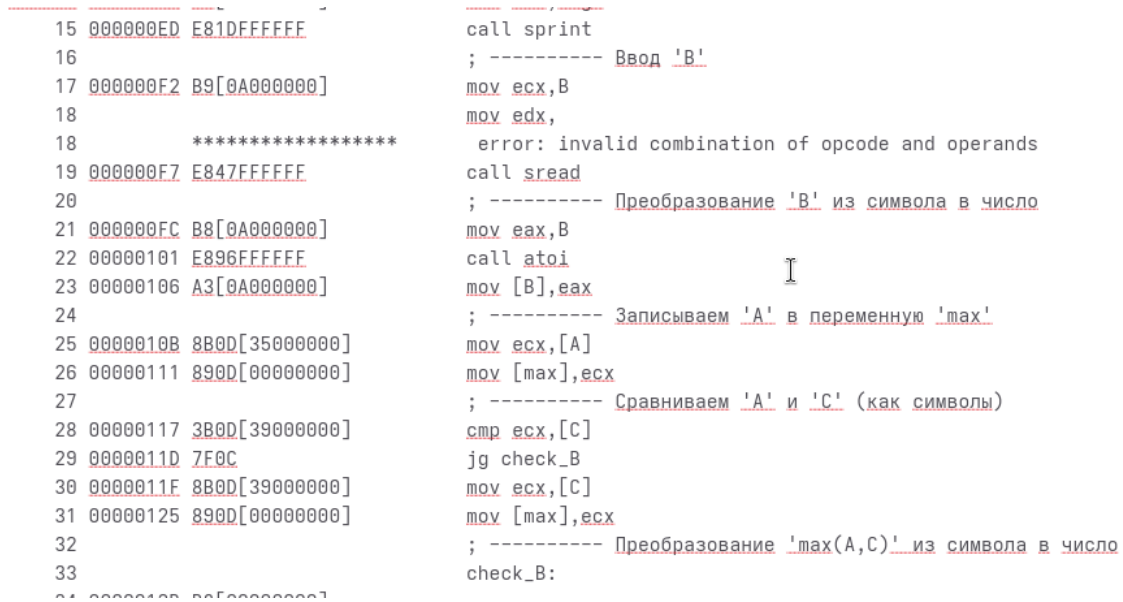


Рисунок 2.11: Файл листинга с ошибкой lab7-2

Объектный файл не смог создаться из-за ошибки. Но получился листинг, где выделено место ошибки.

2.3 Задание для самостоятельной работы

Напишите программу нахождения наименьшей из 3 целочисленных переменных a,b и c. Значения переменных выбрать из табл. 7.5 в соответствии с вариантом, полученным при выполнении лабораторной работы № 6. Создайте исполняемый файл и проверьте его работу

для варианта 10 - 41,62,35

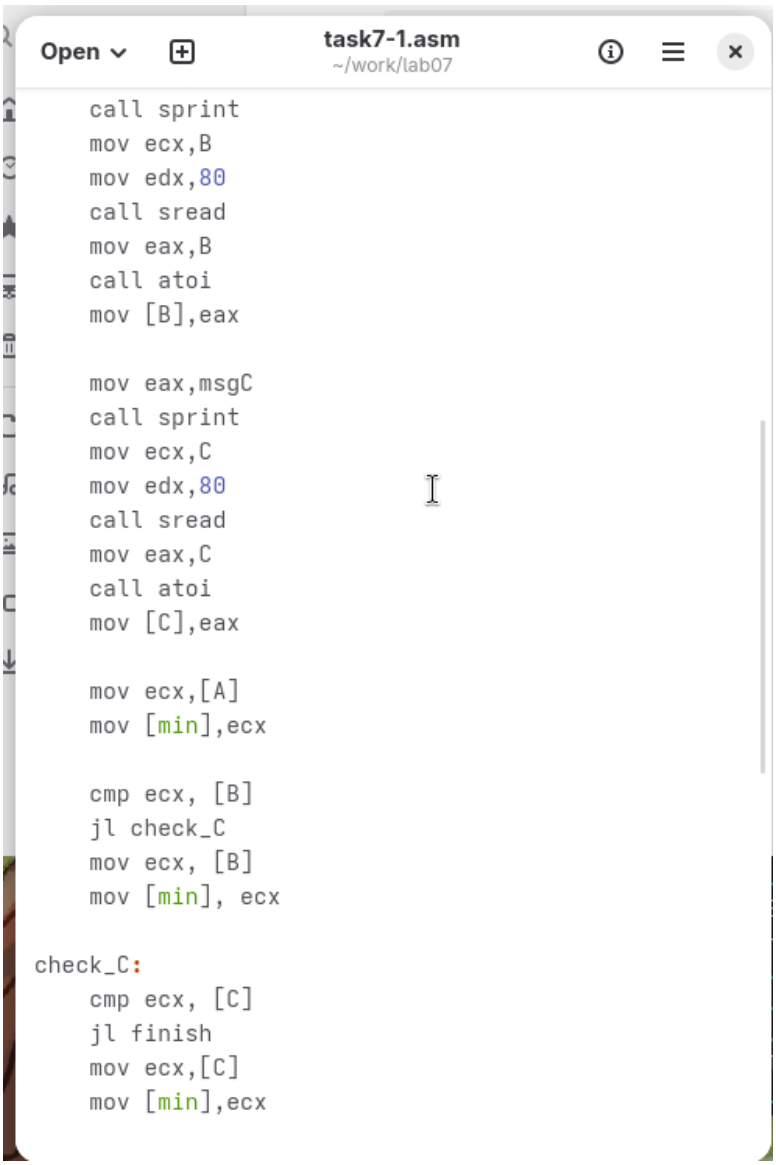


Рисунок 2.12: Программа в файле task7-1.asm

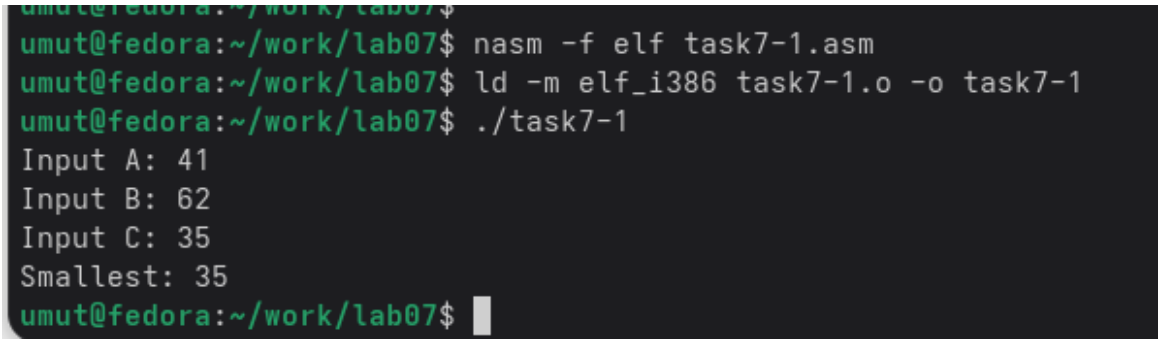


Рисунок 2.13: Запуск программы task7-1.asm

Напишите программу, которая для введенных с клавиатуры значений x и a вычисляет значение заданной функции f(x) и выводит результат вычислений. Вид функции f(x) выбрать из таблицы 7.6 вариантов заданий в соответствии с вариантом, полученным при выполнении лабораторной работы № 7. Создайте исполняемый файл и проверьте его работу для значений X и a из 7.6.

для варианта 10

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

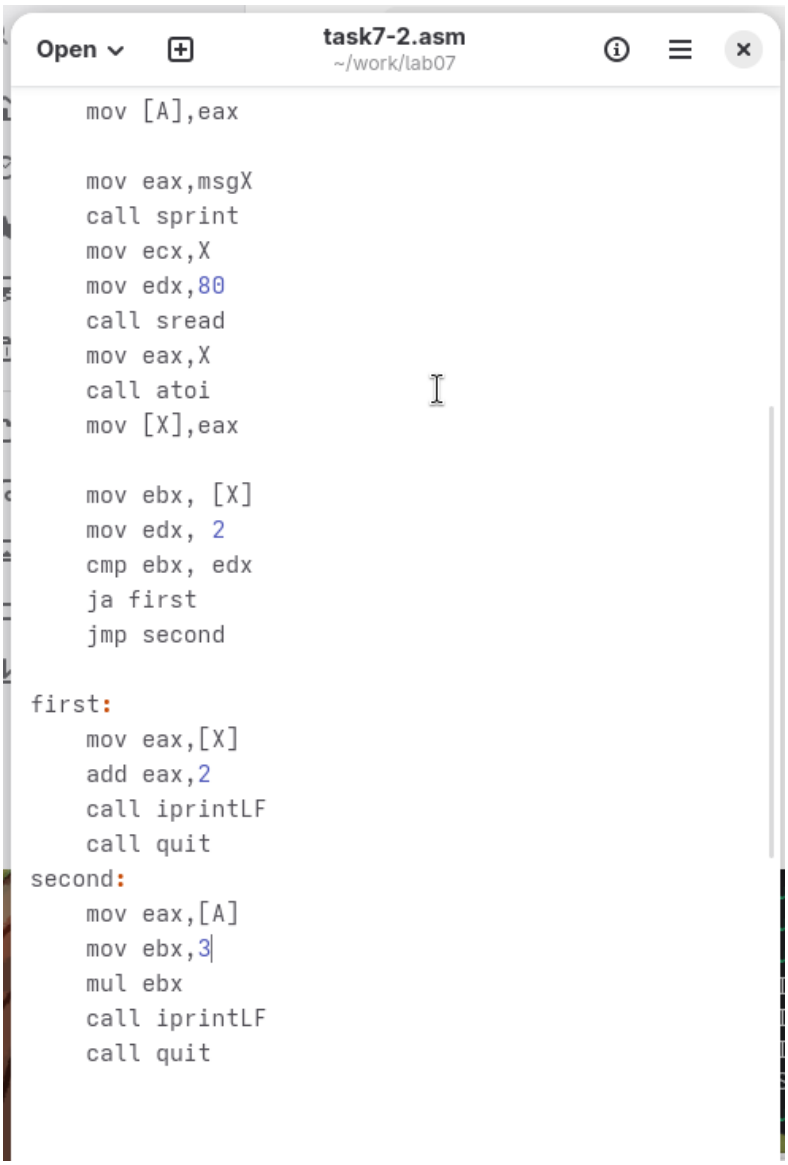


Рисунок 2.14: Программа в файле task7-2.asm

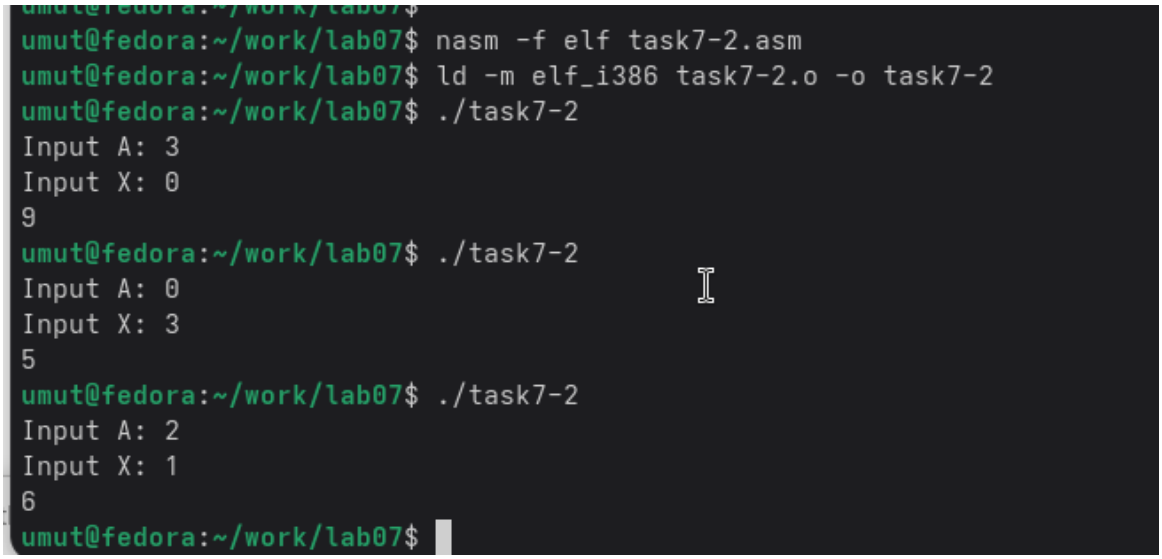


Рисунок 2.15: Запуск программы task7-2.asm

3 Выводы

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