Лабораторная работа №7

Команды безусловного и условного переходов в Nasm. Программирование ветвлений.

Славинский Владислав Вадимович

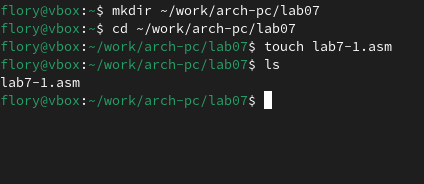
Содержание

# 1 Цель работы

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

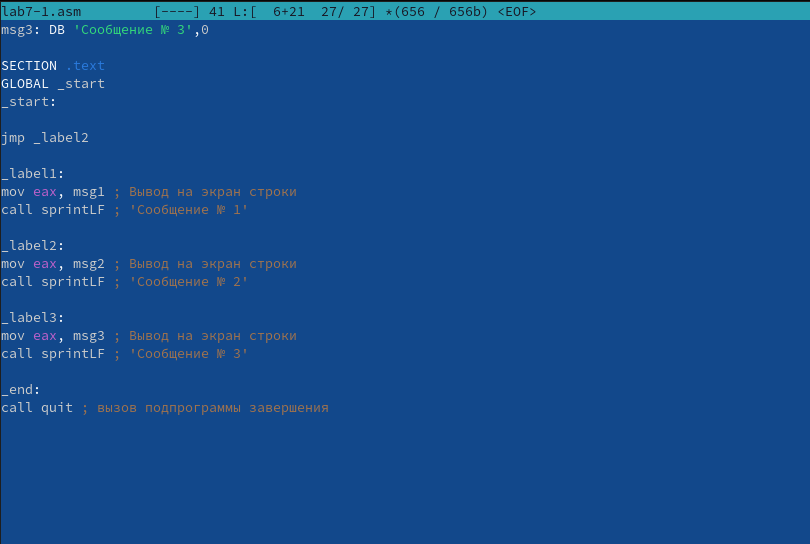
# 2 Выполнение лабораторной работы

1)Создал каталог для лабораторной работы №7 и создал файл lab7-1.asm: (Рис. 2.1)



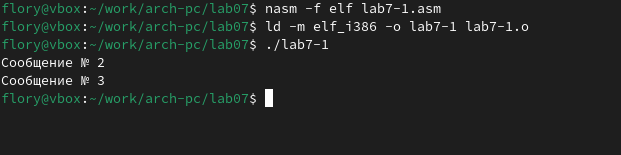
Создание lab7-1

2)Ввел код программы с использованием конструкции jmp: (Рис. 2.2)



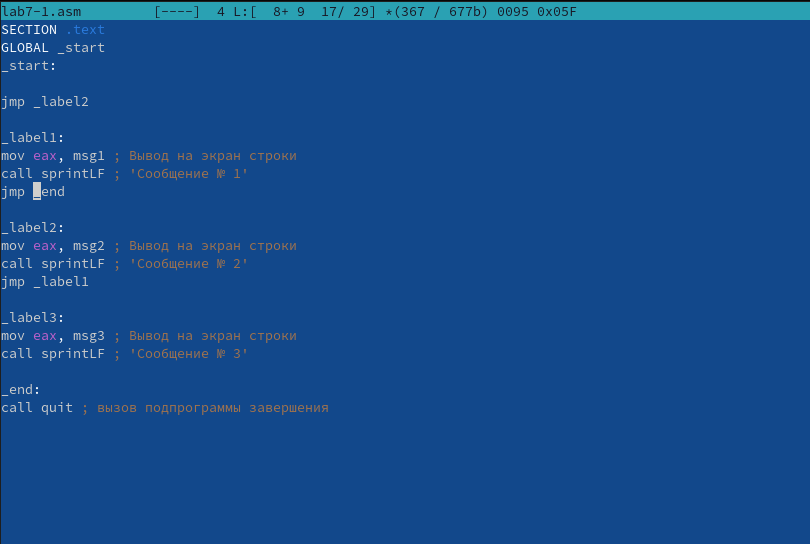
Код программы

3)Создал исполняемый файл и запустил его: (Рис. 2.3)



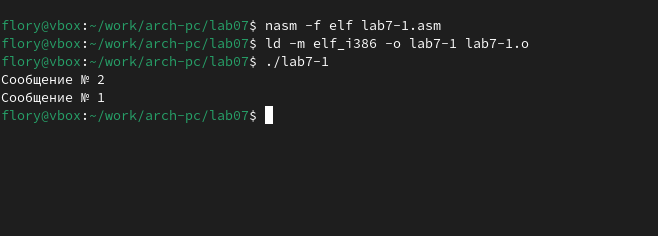
Запуск

4)Теперь изменим программу, чтобы она выводил сначала сообщение № 2, а потом сообщение № 1: (Рис. 2.4)



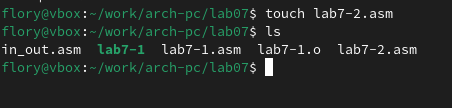
Замена

5)Запустил измененный файл: (Рис. 2.5)



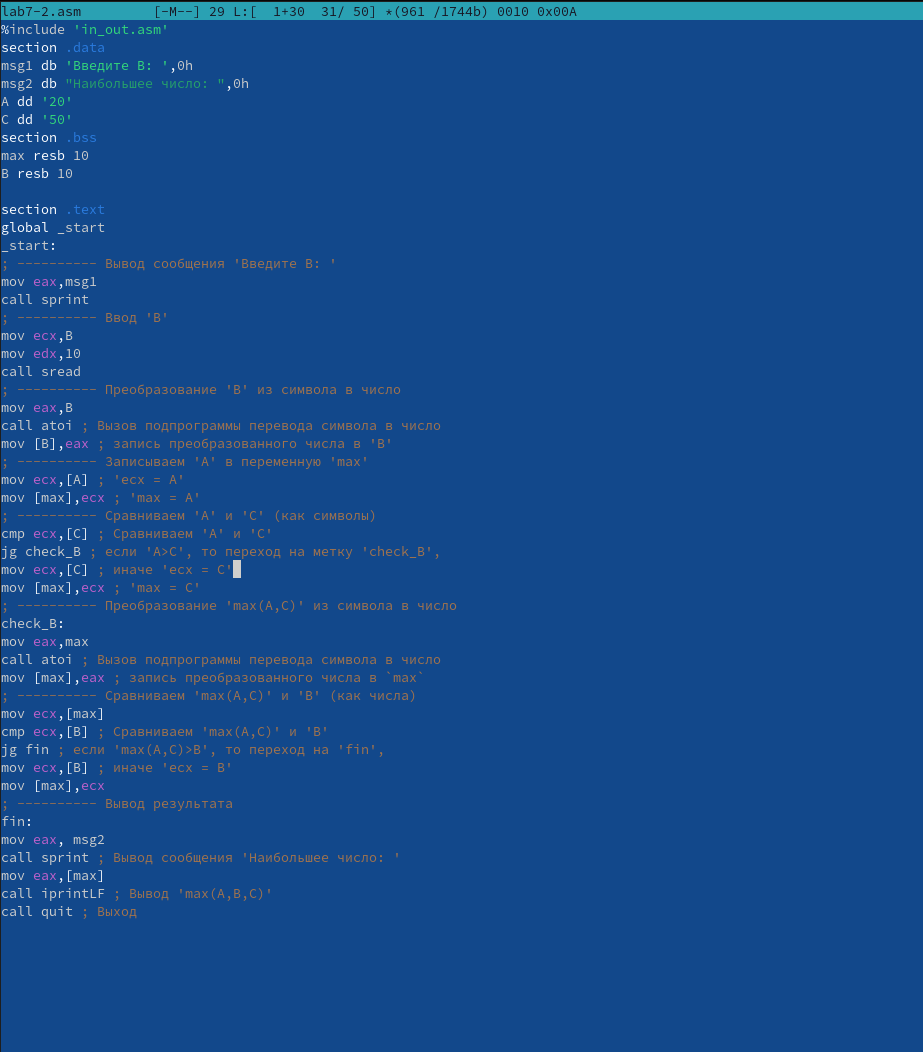
Запуск

6)Создал файл lab7-2.asm: (Рис. 2.6)



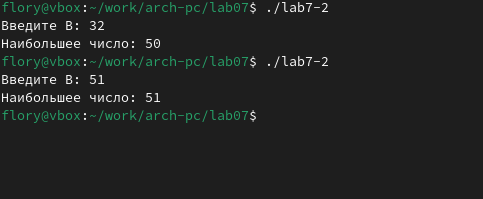
Создание файла lab7-2.asm

7)Ввел код программы, которая определяет и выводит на экран наибольшую из 3 целочисленных переменных: A,B и C:(Рис. 2.7)



Программа

8)Запустил его и убедился в работоспособности:(Рис. 2.8)



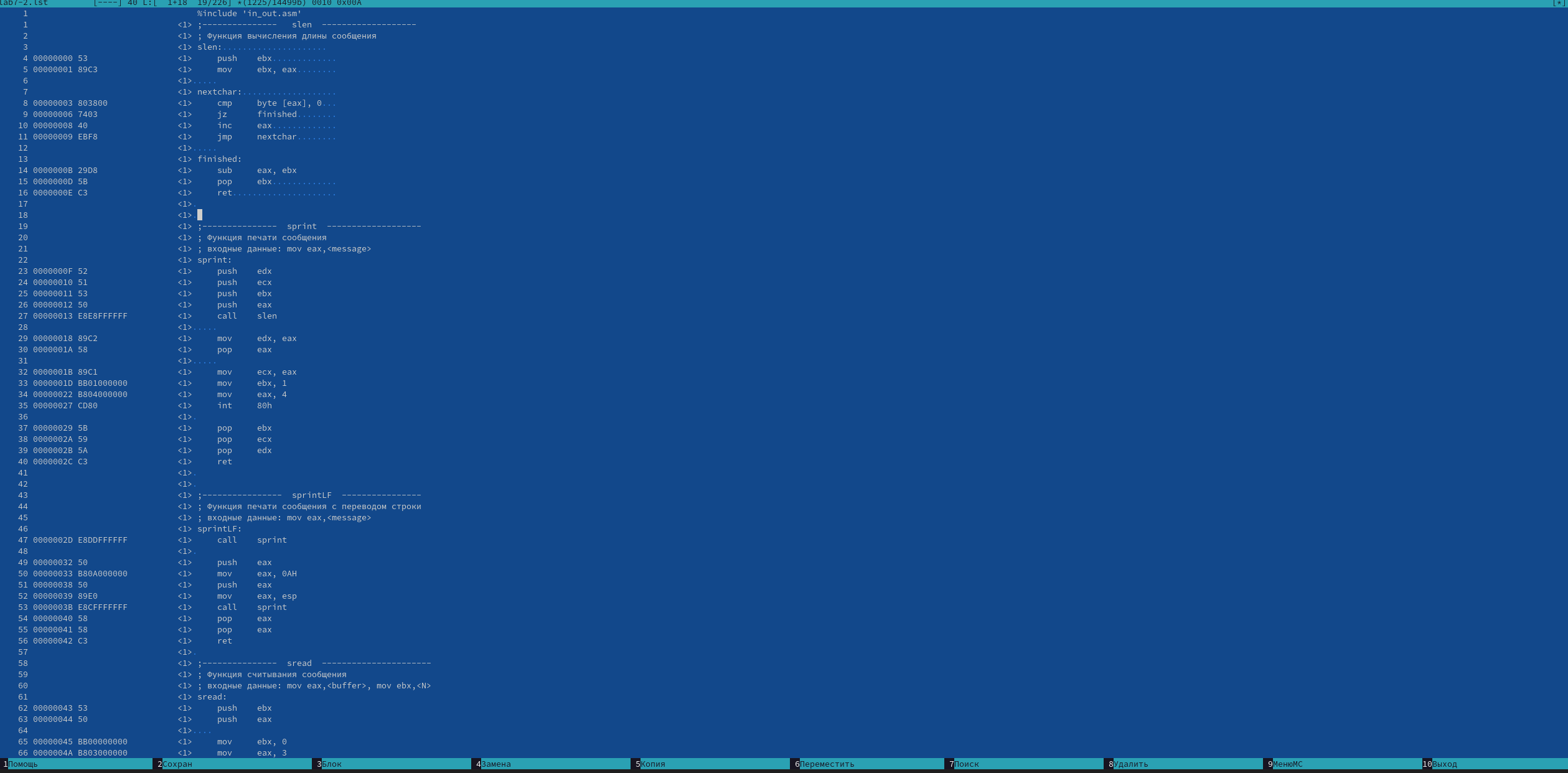
Запуск

9)Получаю файл листинга, указав ключ -l и задав имя файла листинга в командной строке, и открыл файл с помощью редактора:(Рис. 2.9)

Получение файла листинга

Получение файла листинга

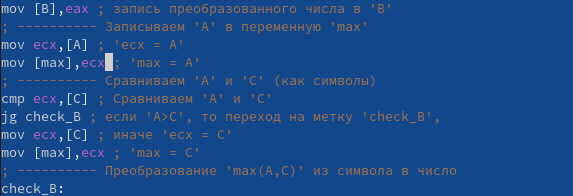
10)Внимательно ознакомился с его форматом и содержимым:(Рис. 2.10)



Содержимое файла

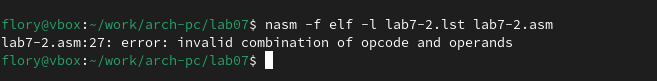
Объясню что делают 8, 9 и 10 строки. Строка 8 cmp byte [eax], 0. Эта инструкция сравнивает байт, находящийся по адресу, на который указывает регистр eax, с нулем. cmp - команда, которая выполняет вычитание, но не сохраняет результат. [eax] указывает на значение по адресу, хранящемуся в eax. Если значение равно нулю, то флаг нуля будет установлен. Строка 9 jz finished. Эта инструкция выполняет переход к метке finished, если установлен флаг нуля(если результат предыдущего сравнения был бы равен нулю). jz - условный переход, который срабатывает, если результат предыдущей операции (в данном случае cmp) указывает на то, что сравниваемые значения равны (т.е. byte [eax] равно 0). Если байт по адресу eax равен 0, выполнение кода переходит к метке finished, что означает конец строки или конец строки в строке. Строка 10: inc eax. Инструкция увеличивает значение в регистре eax на 1. inc увеличивает значение в указанном регистре на 1, eax указывает на текущий символ в строки, и эта команда перемещает указатель к следующему символу. Если текущий символ не равен 0, то eax будет увеличен, чтобы указать на следующий символ в строке.

11)Удалю одну операнду ecx в строке mov [max], ecx :(Рис. 2.11)



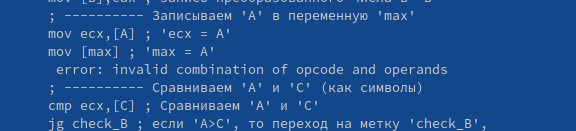
Вычисление выражения

12)Создал исполняемый файл и запустил его. Вывод: ошибка:(Рис. 2.12)



Запуск вычисления выражения

13)Открою теперь файл листинга и проверим его на ошибку:(Рис. 2.13)

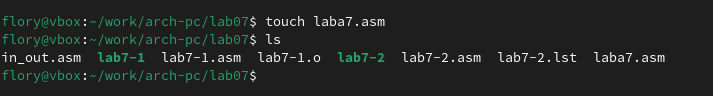


Запуск вычисления выражения

В листинге отображается ошибка

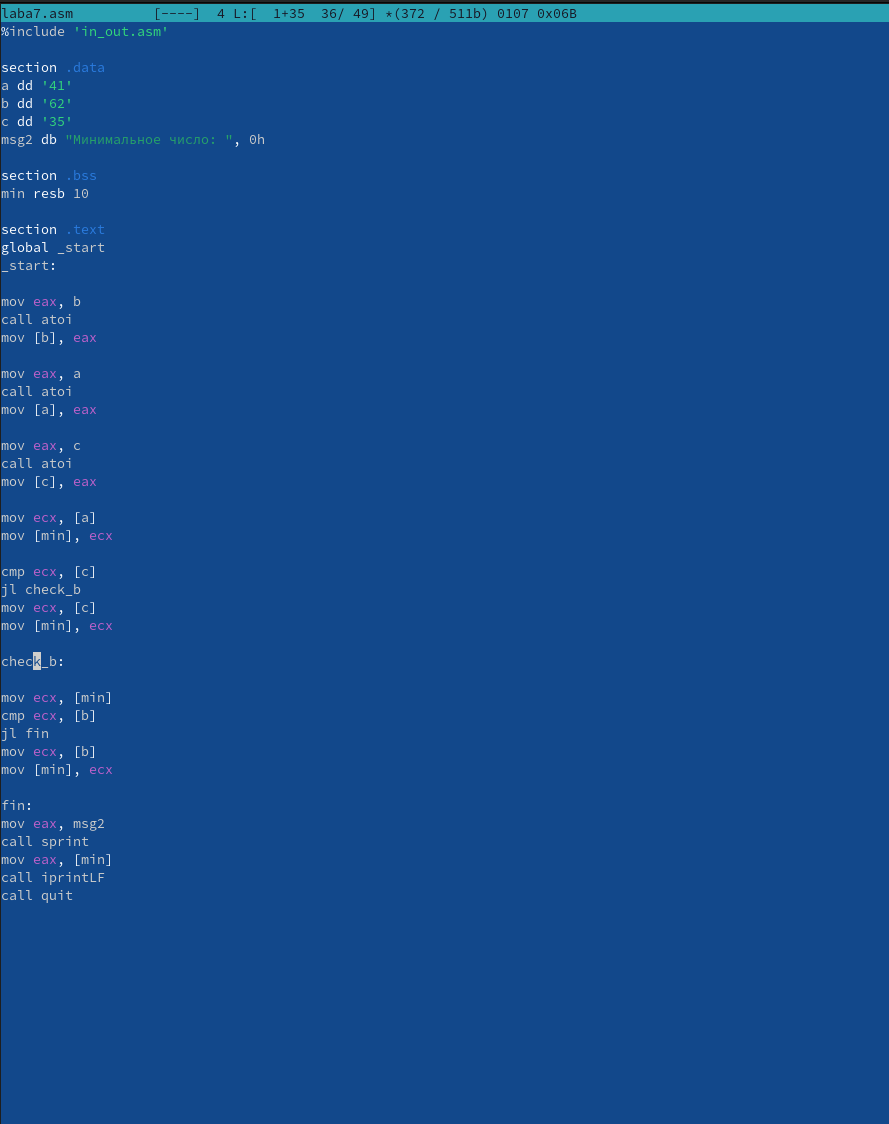
# 3 Выполнение задания для самостоятельной работы

1)Создам первый файл laba7.asm для самостоятельной работы. Вариант 10. :(Рис. 3.1)



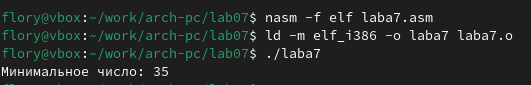
Создание laba7.asm

2)Написал код программы для нахождения наименьшей из 3 целочисленных переменных.(Рис. 3.2)



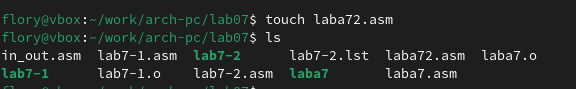
Код программы для нахождения наименьшего

3)Создал исполняемый файл и запустил его.(Рис. 3.3)



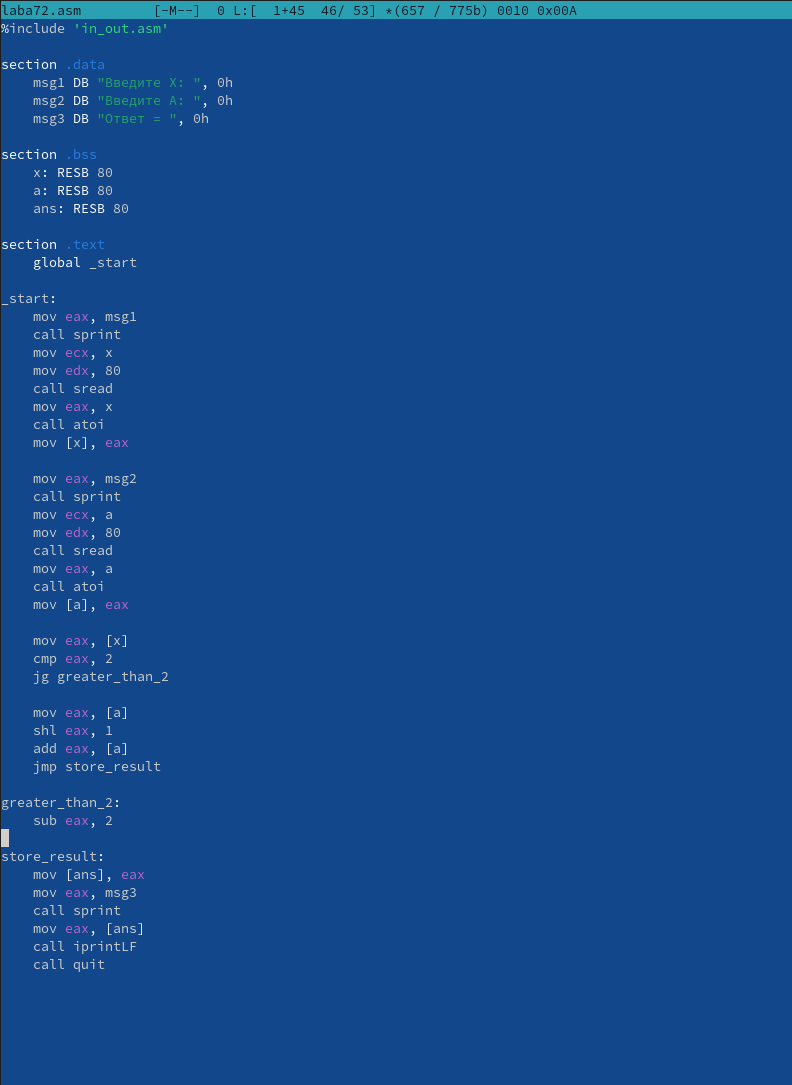
Запуск программы

1. Создам второй файл laba72.asm для второго задания. Рис. 3.4)



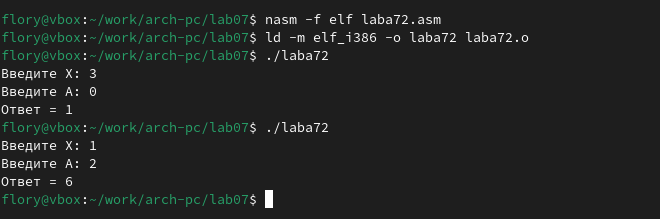
Создание laba72.asm

1. Написал код программы для вычисления значения заданной функции f(x). (Рис. 3.5)



Код программы

1. Создал исполняемый файл и запустил его. (Рис. 3.6)



Запуск программы

# 4 Вывод

В ходе выполнения лабораторной работы я изучил команды условного и безусловного переходов, также приобрел навыки написания программ с использованием переходов. Ознакомился с назначением и структурой файла листинга.