Лабораторная работа №7

Команды безусловного и условного переходов в Nasm. Программирование ветвлений.

Баранов Никита Дмитриевич

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 Задание

Написать программы для решения системы выражений.

# 3 Выполнение лабораторной работы

Создайте каталог для программам лабораторной работы № 7, перейдите в него и создайте файл lab7-1.asm. Инструкция jmp в NASM используется для реализации безусловных переходов. Рассмотрим пример программы с использованием инструкции jmp. Введите в файл lab7-1.asm текст программы из листинга 7.1.Создайте исполняемый файл и запустите его(рис. fig. 1)(рис. fig. 2)(рис. fig. 3).

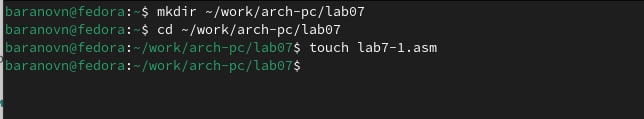


Рис. 1: Создаем каталог и файл .asm

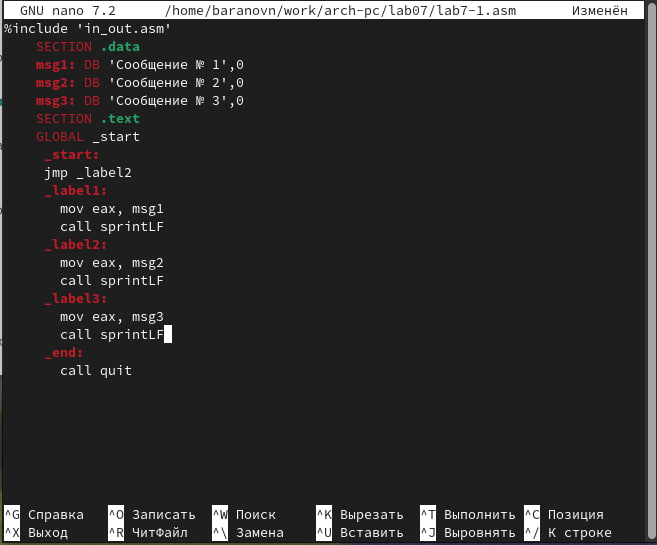


Рис. 2: Вводим программу в файл

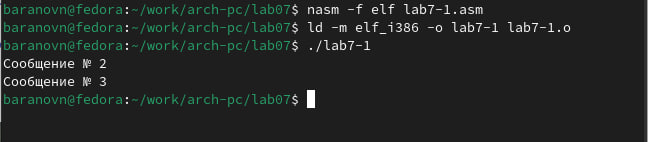


Рис. 3: Создаем исполняемый файл и запускаем его

Наш вывод совпал с выводом в инструкции.Изменим программу таким образом, чтобы она выводила сначала ‘Сообщение № 2’, потом ‘Сообщение № 1’ и завершала работу.Создайте исполняемый файл и проверьте его работу. Измените текст программы добавив или изменив инструкции jmp, чтобы вывод программы был Сообщение № 3 Сообщение № 2 Сообщение № 1.(рис. fig. 4)(рис. fig. 5)(рис. fig. 6)(рис. fig. 7)

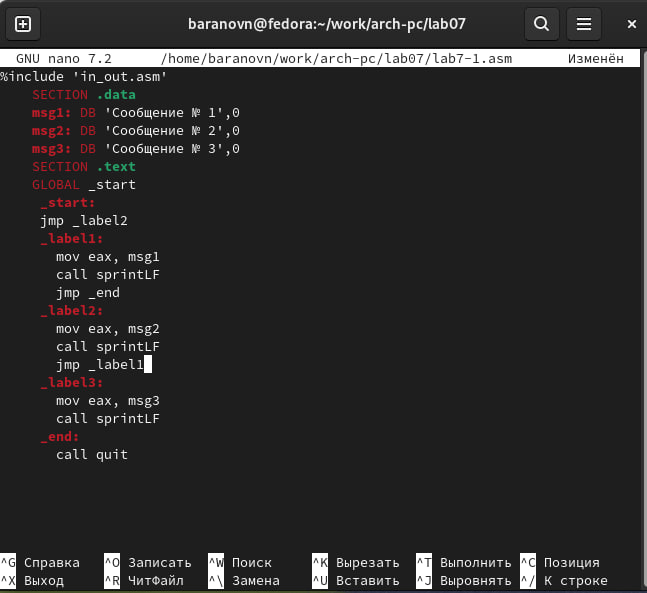


Рис. 4: Изменяем текст программы

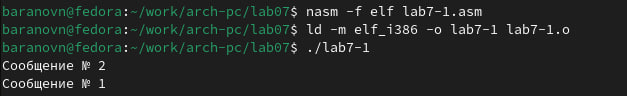


Рис. 5: Создаем исполняемый файл и проверяем его работу

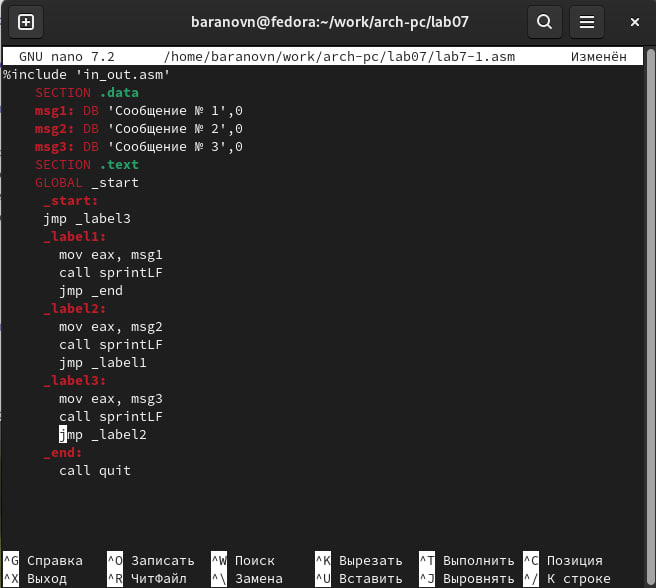


Рис. 6: Редактируем текст программы для новых условий

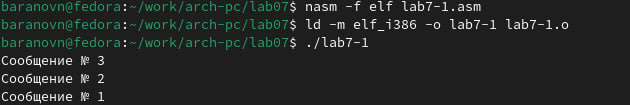


Рис. 7: Создаем исполняемый файл и проверяем его работу

Создайте файл lab7-2.asm в каталоге ~/work/arch-pc/lab07. Внимательно изучите текст программы из листинга 7.3 и введите в lab7-2.asm(рис. fig. 8)(рис. fig. 9)(рис. fig. 10)

Создаем файл lab7-2.asm

Рис. 8: Создаем файл lab7-2.asm

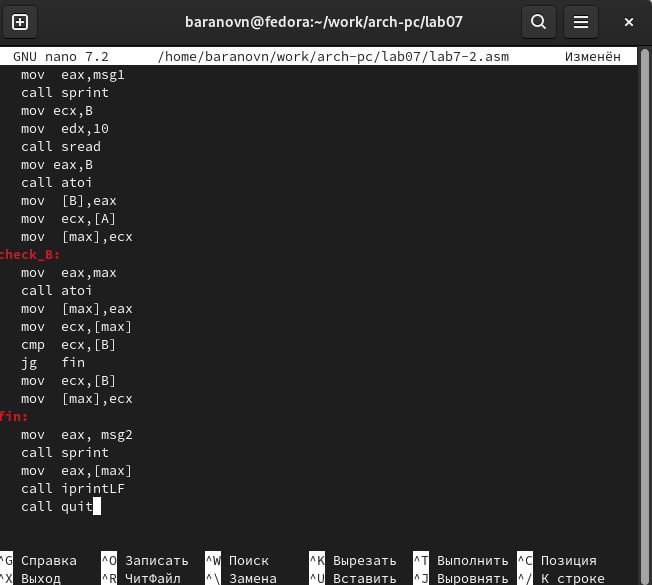


Рис. 9: Записываем программу

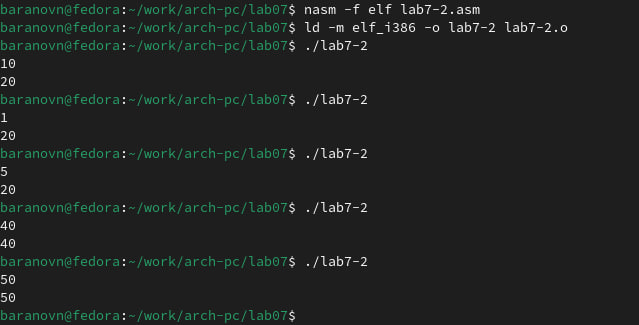


Рис. 10: Создаем исполняемый файл и проверяем работу

Создайте файл листинга для программы из файла lab7-2.asm. Откройте файл листинга lab7-2.lst с помощью любого текстового редактора.Внимательно ознакомиться с его форматом и содержимым. Подробно объяснить содержимое трёх строк файла листинга по выбору. Откройте файл с программой lab7-2.asm и в любой инструкции с двумя операндами удалить один операнд. Выполните трансляцию с получением файла листинга.Какие выходные файлы создаются в этом случае? Что добавляется в листинге?(рис. fig. 11)(рис. fig. 12)

Создаем файл листинга и открываем в текстовом редакторе

Рис. 11: Создаем файл листинга и открываем в текстовом редакторе

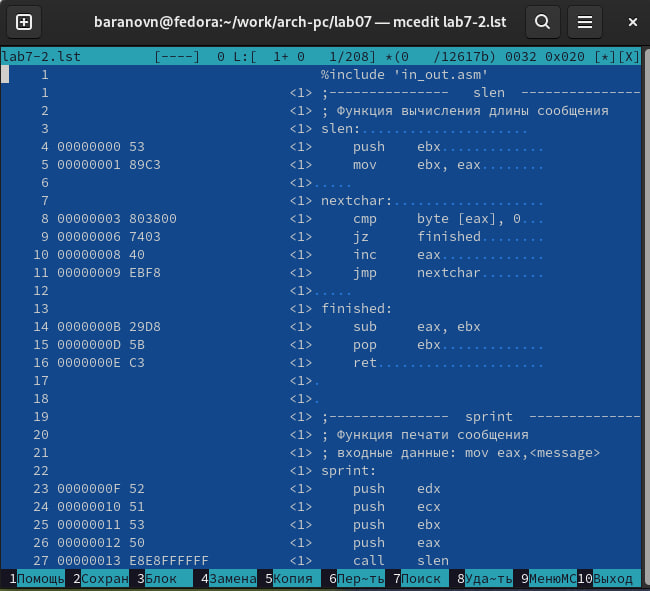


Рис. 12: Открываем файл и ознакамливаемся с форматом

Строка 5 - 00000001 89C3 mov ebx, eax 00000001 - адрес в сегменте кода 89C3 - машинный код для инструкции mov ebx,eax - присваивание переменной ebx значения, хранящееся в регистре eax Строка 26 - 00000012 50 push eax 00000012 - адрес в сегменте кода 50 - машинный код для инструкции push eax - значение из регистра eax помещается в стек Строка 53 - 0000003B E8CFFFFFFF call sprint 0000003B - адрес в сегменте кода E8CFFFFFFF - машинный код для инструкции call sprint - вызов функции sprint, которая выводит данные на экран

# 4 Самостоятельная работа

Напишите программу нахождения наименьшей из 3 целочисленных переменных a, b, c. Значения переменных выбрать из табл. 7.5 в соответствии с вариантом, полученным при выполнении лабораторной работы № 7. Создайте исполняемый файл и проверьте его работу(рис. fig. 13)(рис. fig. 14)

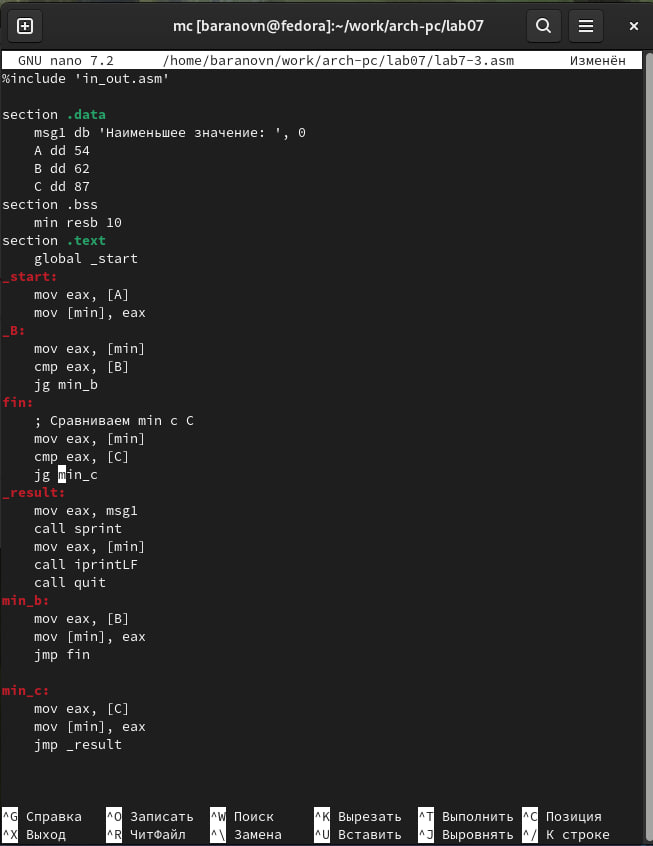


Рис. 13: Пишем программу для определения минимального значения

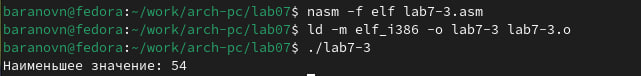


Рис. 14: Создаем исполняемый файл и проверяем работу для значений 54,62,87

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.(рис. fig. 15)(рис. fig. 16)

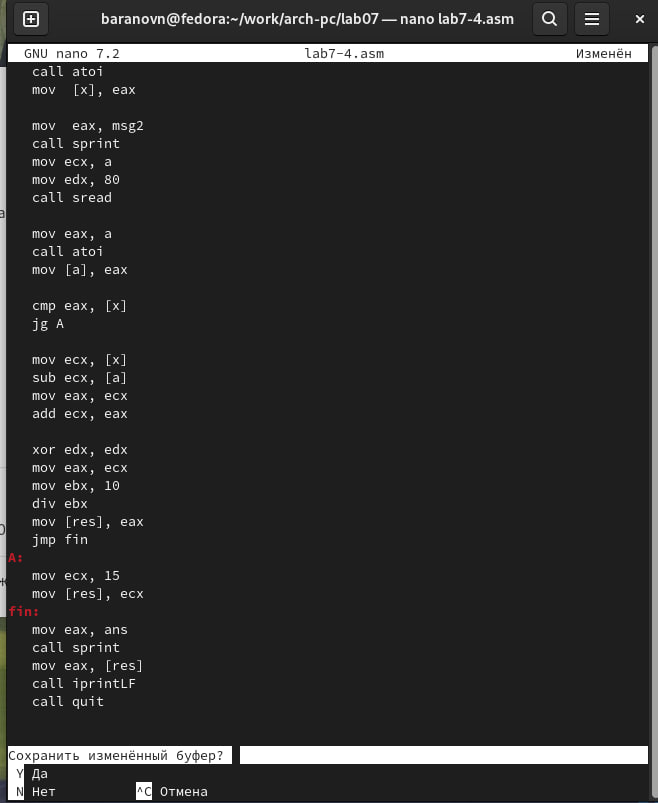


Рис. 15: Пишем программу для нашей функции- номер 5

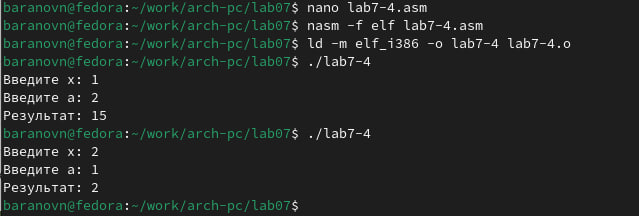


Рис. 16: Создаем объектный файл и проверяем его работу на двух данных нам примерах

# 5 Выводы

Мы познакомились с структурой файла листинга, изучили команды условного и безусловного перехода.