Лабораторная работа №7

Команды безусловного и условного переходов в Nasm. Программирование ветвлений.

Самарханова Полина Тимуровна

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

1.Реализация переходов в NASM 2.Изучение структуры файлов листинга 3.Задания для самостоятельной работы

# 3 Теоретическое введение

Для реализации ветвлений в ассемблере используются так называемые команды передачи управления или команды перехода. Можно выделить 2 типа переходов: • условный переход – выполнение или не выполнение перехода в определенную точку программы в зависимости от проверки условия. • безусловный переход – выполнение передачи управления в определенную точку про- граммы без каких-либо условий. Адрес перехода может быть либо меткой, либо адресом области памяти, в которую предва- рительно помещен указатель перехода. Кроме того, в качестве операнда можно использовать имя регистра, в таком случае переход будет осуществляться по адресу, хранящемуся в этом регистре. Флаг – это бит, принимающий значение 1 («флаг установлен»), если выполнено некоторое условие, и значение 0 («флаг сброшен») в противном случае. Флаги работают независимо друг от друга, и лишь для удобства они помещены в единый регистр — регистр флагов, отра- жающий текущее состояние процессора. Инструкция cmp является одной из инструкций, которая позволяет сравнить операнды и выставляет флаги в зависимости от результата сравнения. Инструкция cmp является командой сравнения двух операндов и имеет такой же формат, как и команда вычитания

# 4 Выполнение лабораторной работы

## 4.1 Реализация переходов в NASM

Я создала директорию lab07 в каталоге,в котором работал на протяжении предыдущих работ. После чего создаюла новый файл lab7-1.asm, чтобы далее записать в него код программы (рис. ??).

Создание каталога и файла

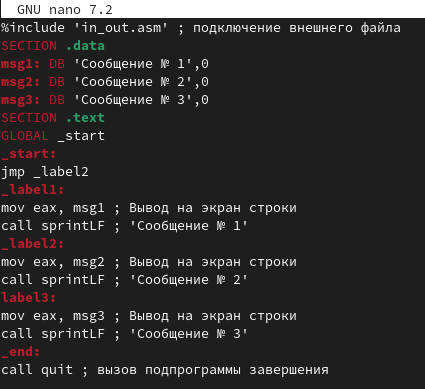
Создание каталога и файла

Не забываю также скопировать в созданный каталог файл in\_out.asm, так как далее он будет использоваться во всех программах (рис. ??).

Копирование файла

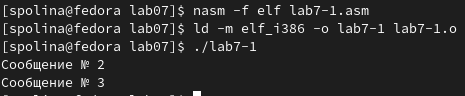
Копирование файла

Открыла файл lab7-1.asm в GNU nano и вставила текст программы из листинга 1 (рис. ??).



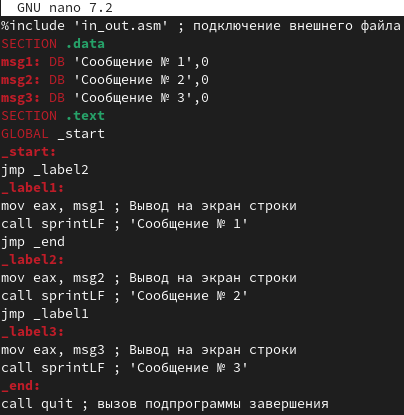
Редактирование программы

После чего я создала объектный файл программы,кампановала его и запустила код (рис. ??). Благодаря команде jmp программа сразу перепрыгивает ко второму действию,игнорируя первый этап кода.



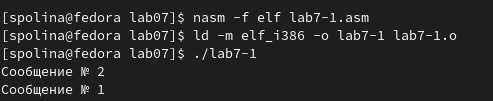
Запуск кода

Далее я изменила текст программы в сооитветствии с листингом 2 (рис. ??).



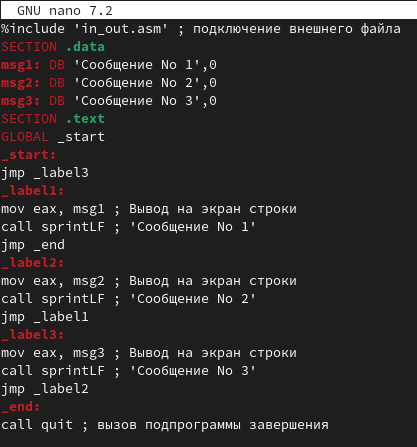
Редактирование программы

Кампановала созданный объектный файл и запустила программу (рис. ??)



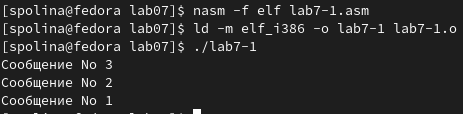
Запуск программы

Редактирую код программы в соответствии с заданием: сначала выводится “Сообщение №3”, затем “Сообщение №2”, затем “Сообщение №1”, для этого я использую всё ту же команду jmp (рис. ??).



Создание новой программы

Далее я запустила программу и проверила правильность написания программы (рис. ??). После я запусила программу, она выдаёт результат в правильном порядке.



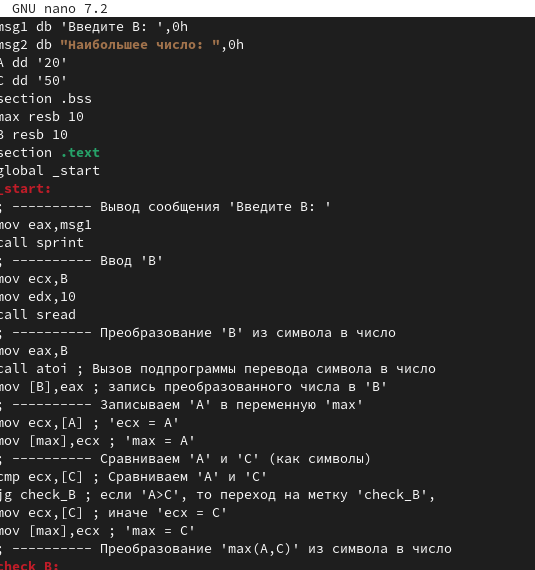
Запуск программы

Я создала новый файл lab7-2.asm,используя утилиту touch (рис. ??).

Создание файла

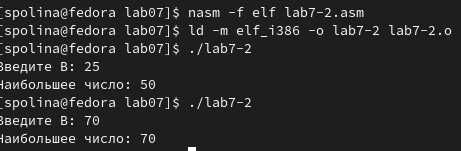
Создание файла

Вставила в созданный файл текст кода,скопировав листинг 7.3 (рис. ??).



Редактирование программы

Я запустила код дважды, в первый раз я ввела число,которое меньше одной из констант, а во второй-больше (рис. ??).В обоих случаях программа выдает наибольшее из трёх чисел.



Запуск программы

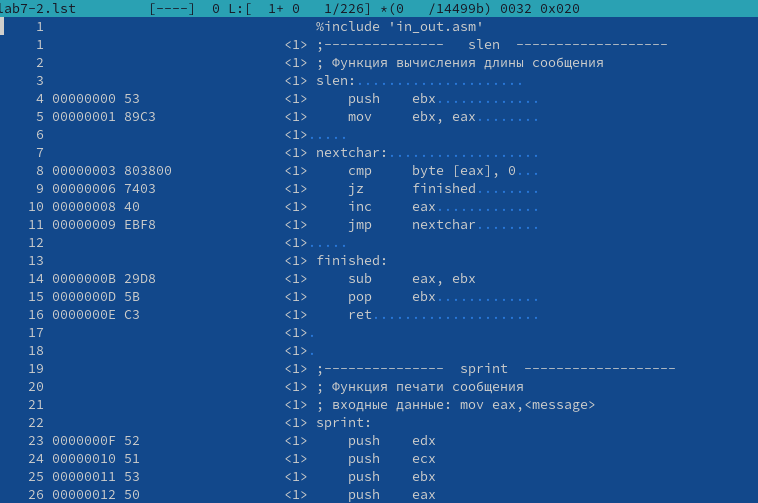
## 4.2 Изучение структуры файлов листинга

Я создала файл листинга для lab7-2.asm (рис. ??).

Создание файла листинга

Создание файла листинга

Далее я открыла созданный файл с помощью mcedit,используя команду “mcedit lab7-2.lst” (рис. ??).



Файл листинга

Первая строка,которую я хочу подробно описать,является строка №5 (рис. ??). Первое число в строке как раз и указывает на номер строки, после чего мы можем увидеть адрес данной строки, далее у нас идет машинный код,в который ассемблируется инструкция.То есть инструкция “mov ebx,eax” ассемблируется в 89С3. В данном случае, 89С3-это инструкция на машинном языке по вызову и присваиванию регистра. После этого мы можем увидеть исходный текст программы.

Первая строка

Первая строка

Вторая строка,которую я хочу подробно описать,является строка №11 (рис. ??). Первое число в строке как раз и указывает на номер строки, после чего мы можем увидеть адрес данной строки, далее у нас идет машинный код,в который ассемблируется инструкция.То есть инструкция “jmp nextchar” ассемблируется в EBF8. В данном случае, EBF8-это инструкция на машинном языке по переходу на другую строку. После этого(в правой части) мы можем увидеть исходный текст программы.

Вторая строка

Вторая строка

Третья строка,которую я хочу подробно описать,является строка №39 (рис. ??). Первое число в строке(самое левое) как раз и указывает на номер строки, после чего мы можем увидеть адрес данной строки, далее у нас идет машинный код,в который ассемблируется инструкция.То есть инструкция “cmp ecx” ассемблируется в 380D. В данном случае, 380D-это инструкция на машинном языке по сравнению чисел. В квадратных скобках рядом можно увидеть адрес,который указывает на значение числа B. После этого(в правой части) мы можем увидеть исходный текст программы.

Третья строка

Третья строка

Далее я открыла файл с программой lab7-2.asm и в инструкции с двумя операндами удалила один из них (рис. ??).

Удаление операнда

Удаление операнда

Далее выполнила трансляцию с получением файла листинга (рис. ??). Транслятор выводит ошибку при ассемблировании,даже указывая на номер строки, и файл листинга не создается.

Попытка создать файл листинга

Попытка создать файл листинга

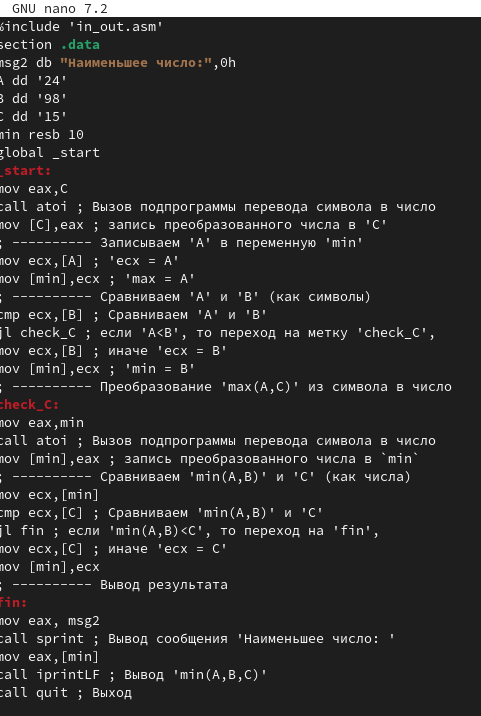
## 4.3 Задания для самостоятельной работы

Я создала файл,в котором делала первое самостоятельное задание (рис. ??).

Создание файла

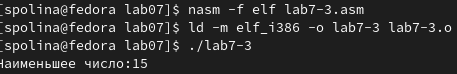
Создание файла

В лабораторной работе №6 мне выпал 9 вариант.Я написала программу,которая выбирает наименьшее число из трех заданных чисел.С начала выбираю меньшее из A и B, а после сравниваю его с C (рис. ??).



Редактирование программы

Запустила, проверила ответ и убедилась,что программа выдает наименьшее из трёх чисел (рис. ??).



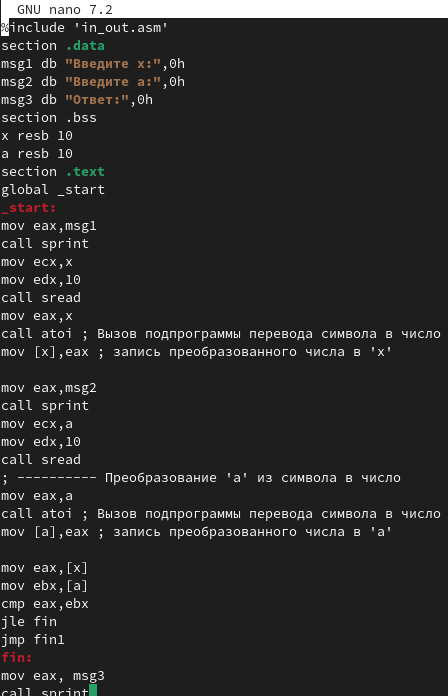
Запуск программы

Для следующего задания создала ещё один файл lab7-4.asm (рис. ??).

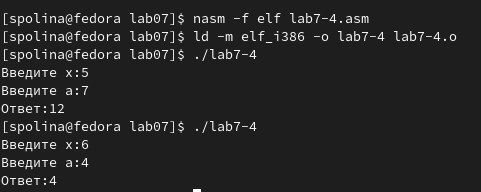
Создание файла

Создание файла

Написала программу по заданию 9 варианта (рис. ??).



Редактирование программы

Компаную и запускаю исполняемый файл,вводя числа для первой проверки (рис. ??). 

# 5 Выводы

В этой работе я научилась работать с переходами в NASM, разобралась со структурой листинговых файлов и научилась применять эти знания для написания программ.

# 6 Список литературы

Лабораторная работа №7.