Отчёт по лабораторной работе № 5

Создание и процесс обработки программ на языке ассемблера NASM

Давит Оганнисян Багратович

Содержание

# 1 Цель работы

Освоение процедуры компиляции и сборки программ, написанных на ассемблере NASM.

# 2 Задание

## 2.1 Программа Hello world!

## 2.2 Транслятор NASM

## 2.3 Расширенный синтаксис командной строки NASM

## 2.4 Компоновщик LD

## 2.5 Запуск исполняемого файла

## 2.6 Задание для самостоятельной работы

# 3 Теоретическое введение

NASM — это открытый проект ассемблера, версии которого доступны под различные операционные системы и который позволяет получать объектные файлы для этих систем. В NASM используется Intel-синтаксис и поддерживаются инструкции x86-64. Типичный формат записи команд NASM имеет вид: [метка:] мнемокод [операнд {, операнд}] [; комментарий] Здесь мнемокод — непосредственно мнемоника инструкции процессору, ко- торая является обязательной частью команды. Операндами могут быть числа, данные, адреса регистров или адреса оперативной памяти. Метка — это иденти- фикатор, с которым ассемблер ассоциирует некоторое число, чаще всего адрес в памяти. Т.о. метка перед командой связана с адресом данной команды. Допустимыми символами в метках являются буквы, цифры, а также следую- щие символы: *, $, #, @,~,. и ?. Начинаться метка или идентификатор могут с буквы, .,*  и ?. Перед иденти- фикаторами, которые пишутся как зарезервированные слова, нужно писать $, чтобы компилятор трактовал его верно (так называемое экранирование). Макси- мальная длина идентификатора 4095 символов. Программа на языке ассемблера также может содержать директивы — ин- струкции, не переводящиеся непосредственно в машинные команды, а управ- ляющие работой транслятора. Например, директивы используются для опреде- ления данных (констант и переменных) и обычно пишутся большими буквами.

# 4 Выполнение лабораторной работы

## 4.1 Создаем каталог для работы с программами на языке ассемблера NASM, переходим в него и создаем текстовый файл с именем hello.asm. Открываем этот файл с помощью любого текстового редактора, например, gedit и вводим нужный нам текст

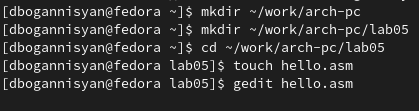


Рис. 1: Создание hello.asm

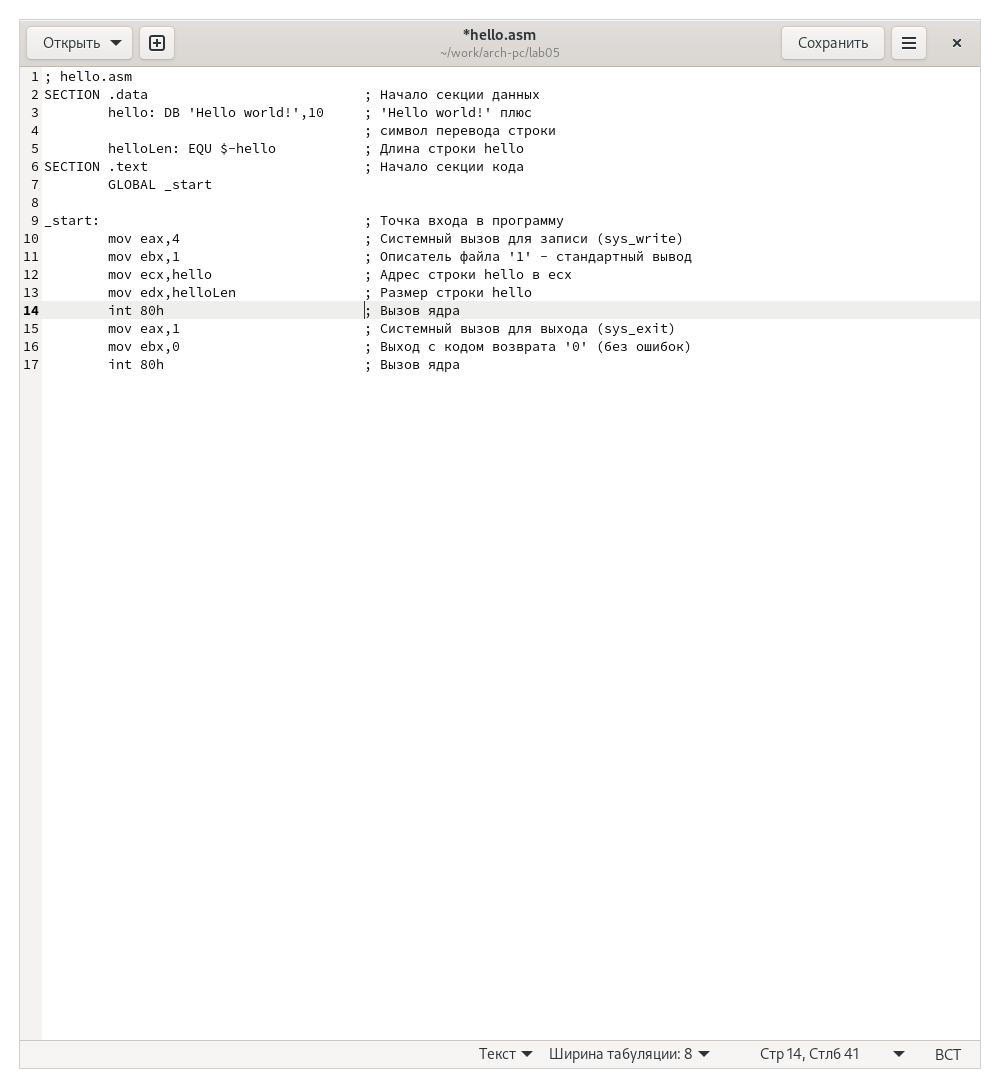


Рис. 2: Текст

## 4.2 Превращение текста программы в объектный код и компилирование hello.asm в obj.o

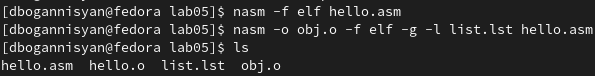


Рис. 3: Превращение текста программы в объектный код и компилирование hello.asm в obj.o

## 4.3 Передача объектного файла на обработку компоновщику

Рис. 4: Передача объектного файла на обработку компоновщику

Рис. 4: Передача объектного файла на обработку компоновщику

## 4.4 Создаем также исполняемый файл main

Рис. 5: Создание также исполняемый файл main

Рис. 5: Создание также исполняемый файл main

## 4.5 Запуск исполняемого файла

Рис. 6: Hello world!

Рис. 6: Hello world!

## 4.6 Задание для самостоятельной работы

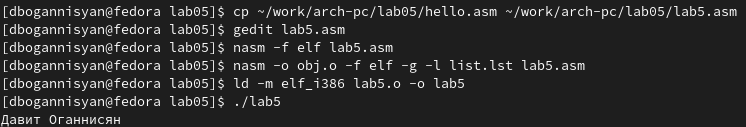


Рис. 7: Фамилия Имя

Рис. 8: Копирование

Рис. 8: Копирование

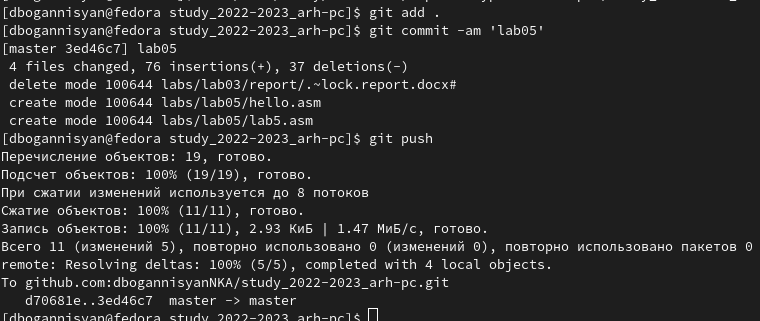


Рис. 9: Загрузка на гитхаб

# 5 Выводы

Я освоил процедуры компиляции и сборки программ, написанных на ассемблере NASM.