Отчет по лабораторной работе №4

Дисциплина: архитектура компьютера

Кузнецова Елизавета Андреевна

Содержание

# 1 Цель работы

Целью работы является освоение процедуры компиляции и сборки программ, написанных на ассамблере NASM.

# 2 Задание

1. Создание программы Hello, world!
2. В каталоге ~/work/arch-pc/lab04 с помощью команды cp создайте копию файла hello.asm с именем lab4.asm
3. С помощью любого текстового редактора внесите изменения в текст программы в файле lab4.asm так, чтобы вместо Hello world! на экран выводилась строка с вашими фамилией и именем.
4. Оттранслируйте полученный текст программы lab4.asm в объектный файл. Выполните компоновку объектного файла и запустите получившийся исполняемый файл.
5. Скопируйте файлы hello.asm и lab4.asm в Ваш локальный репозиторий в ката- лог ~/work/study/2023-2024/“Архитектура компьютера”/arch-pc/labs/lab04/. Загрузите файлы на Github

# 3 Теоретическое введение

Основными функциональными элементами любой электронно-вычислительной машины (ЭВМ) являются центральный процессор, память и периферийные устройства. Взаимодействие этих устройств осуществляется через общую шину, к которой они подключены. Физически шина представляет собой большое количество проводников, соединяющих устройства друг с другом. В современных компьютерах проводники выполнены в виде электропроводящих дорожек на материнской (системной) плате. Основной задачей процессора является обработка информации, а также организация координации всех узлов компьютера. В состав центрального процессора (ЦП) входят следующие устройства: арифметико-логическое устройство (АЛУ) — выполняет логические и арифметические действия, необходимые для обработки информации, хранящейся в памяти; устройство управления (УУ) — обеспечивает управление и контроль всех устройств компьютера; регистры — сверхбыстрая оперативная память небольшого объёма, входящая в состав процессора, для временного хранения промежуточных результатов выполнения инструкций; регистры процессора делятся на два типа: регистры общего назначения и специальные регистры. Для того, чтобы писать программы на ассемблере, необходимо знать, какие регистры процессора существуют и как их можно использовать. Большинство команд в программах написанных на ассемблере используют регистры в качестве операндов. Практически все команды представляют собой преобразование данных хранящихся в регистрах процессора, это например пересылка данных между регистрами или между регистрами и памятью, преобразование (арифметические или логические операции) данных хранящихся в регистрах. Другим важным узлом ЭВМ является оперативное запоминающее устройство (ОЗУ). ОЗУ — это быстродействующее энергозависимое запоминающее устройство, которое на- прямую взаимодействует с узлами процессора, предназначенное для хранения программ и данных, с которыми процессор непосредственно работает в текущий момент. ОЗУ состоит из одинаковых пронумерованных ячеек памяти. Номер ячейки памяти — это адрес хранящихся в ней данных.

# 4 Выполнение лабораторной работы

Создаю каталог для работы с программами на языке ассемблера NASM и с помощью команды cd перемещаюсь в сохданный каталог (рис. [??]).

Создание каталога для работы и перемещение между директориями

Создание каталога для работы и перемещение между директориями

Создаю в текущем каталоге текстовый файл hello.asm с помощью улитилы touch и открываю созданный файл в текстовом редакторе gedit (рис. [??]).

Создание файла и его открытие в текстовом редакторе

Создание файла и его открытие в текстовом редакторе

Заполняю файл, вставляя в него программу (рис. [??]).



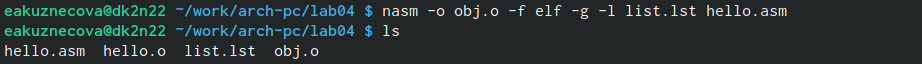
Заполнение файла

Превращаю текст программы в объектный код с помощью транслятора NASM, используя команду nasm -f elf hello.asm. Далее проверяю правильность выполнения команды с помощью команды ls (рис. [??]).

Компиляция текста программы

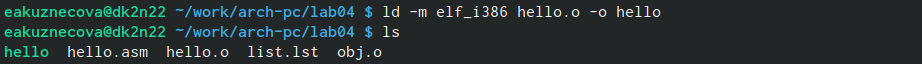
Компиляция текста программы

Использую команду, которая скомпилирует исходный файл hello.asm в obj.o, при этом формат выходного файла будет elf и в него будут включены символы для отладки (ключ -g), кроме того, будет создан файл листинга list.lst.Далее с помощью команды ls проверяю, что файлы были созданы (рис. [??]).



Компиляция текста программы

Передаю объектный файл hello.o на обработку компоновщику LD, чтобы получить исполняемый файл hello. Далее проверяю с помощью команды ls правильность выполнения команды (рис. [??]).



Передача объектного файла на обработку компоновщику

Использую команду, которая позволяет исполняемому файлу иметь имя main, потому что с помощью ключа -o было создано значение main. Объектный файл будет иметь имя obj.o (рис. [??]).

Передача объктного файла на обработку компоновшику

Передача объктного файла на обработку компоновшику

Запускаю на выполнение созданный исполняемый файл hello (рис. [??]).

Запуск исполняемого файла

Запуск исполняемого файла

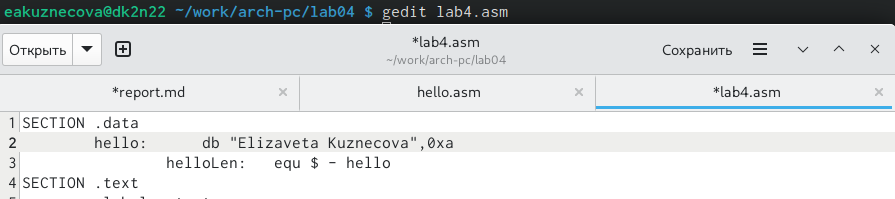
# 5 Выполнение заданий для самостоятельной работы

С помощью команды cp создаю в текущем каталоге копию файла hello.asm с именем lab4.asm (рис. [??]).

Создание копии файла

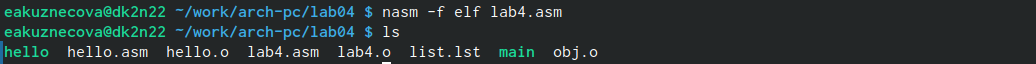
Создание копии файла

С помощью текстового редактора gedit открываю файл lab4.asm, вношу измения в программу, чтобы она выводила мои имя и фамилию (рис. [??]).



Измение программы

Компилирую текст программы в объектный файл. Проверяю с помощью команды ls, что файл lab4.o создан (рис. [??]).



Компиляция текста программы

Передаю объектный файл lab4.o на обработку компоновшику LD, чтобы получить исполняемый файл lab4 (рис. [??]).

Передача объктного файла на обработку компоновщику

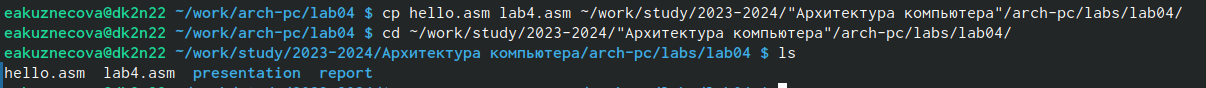
Передача объктного файла на обработку компоновщику

Запускаю исполняемый файл lab4 (рис. [??]).

Запуск исполняемого файла

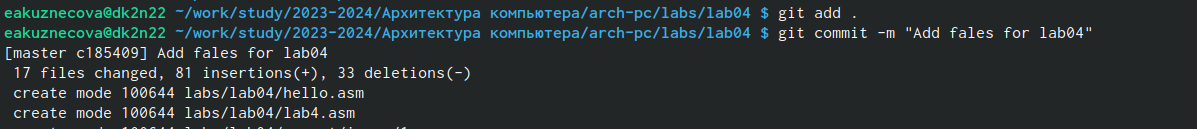
Запуск исполняемого файла

Копирую файлы hello.asm и lab4.asm в свой локальный репозиторий в каталог ~/work/study/2023-2024/“Архитектура компьютера”/arch-pc/labs/lab04/. Перехожу в этот каталог и проверяю наличие файлов в нем (рис. [??]).



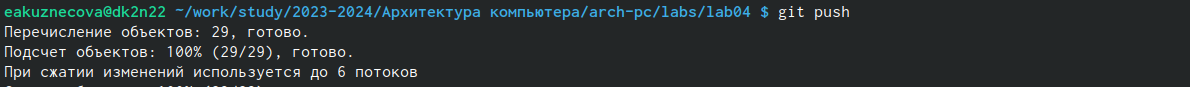
Загрузка файлов на локальный репозиторий

С помощью команд git add . и git commit добавляю файлы на GitHub (рис. [??]).



Добавление файлов на GitHub

Отправляю файлы на сервер с помощью команды git push (рис. [??]).



Отправка файлов

# 6 Выводы

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