Oтчёта по лабораторной работе 4

НММбд-04-24

Ракутуманандзара Цантамписедрана Сарубиди

Содержание

Список иллюстраций

Список таблиц

# 1 Цель работы

Целью данной лабораторной работы является освоение процедуры оформления отчетов с помощью легковесного языка разметки Markdown.

# 2 Задание

1. Установка необходимого ПО  
2. Заполнение отчета по выполнению лабораторной работы №4 с помощью языка разметки Markdown  
3. Задание для самостоятельной работы

# 3 Теоретическое введение

Markdown - легковесный язык разметки, созданный с целью обозначения форматирования в простом тексте, с максимальным сохранением его читаемости человеком, и пригодный для машинного преобразования в языки для продвинутых публикаций. Внутритекстовые формулы делаются аналогично формулам LaTeX. В Markdown вставить изображение в документ можно с помощью непосредственного указания адреса изображения. Синтаксис Markdown для встроенной ссылки состоит из части [link text], представляющей текст гиперссылки, и части (file-name.md) – URL-адреса или имени файла, на который дается ссылка. Markdown поддерживает как встраивание фрагментов кода в предложение, так и их размещение между предложениями в виде отдельных огражденных блоков. Огражденные блоки кода — это простой способ выделить синтаксис для фрагментов кода.

# 4 Выполнение лабораторной работы

##1 Создание программы Hello world!

я буду использовать mkdir для создания указанного каталога, а затем с помощью команды cd войду в каталог, в котором буду работать, там я создам пустой текстовый файл «hello.asm» с помощью команды touch (рис 1).

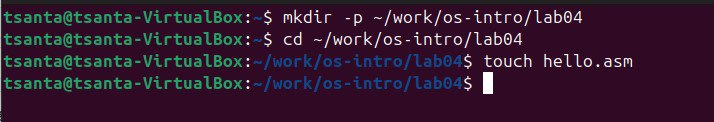


Рис. 1: рис 1

Я открываю созданный файл в текстовом редакторе и заполню файл, вставив в него программу вывода «Hello word!» (рис 2).

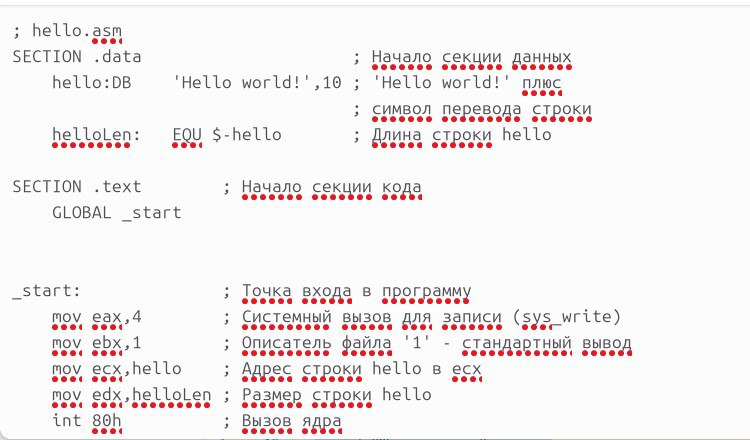


Рис. 2: рис 2

##2 Транслятором NASM

Я преобразую текст программы в вывод «Hello world!» в объектный код с помощью транслятора NASM с помощью команды nasm -f elf hello.asm. Далее проверяю правильность выполнения команды с помощью команды ls и, как показано, файл hello.o создан (рис3).

Рис. 3: рис3

Рис. 3: рис3

##3 Расширенным синтаксисом командной строки NASM

Я введу команду, которая скомпилирует файл hello.asm в файл obj.o, и файл будет содержать символы отладки (переключатель -g), также используя переключатель -l будет создан файл листинга list.lst. Далее проверяю с помощью команды ls правильность выполнения команды(рис 4)

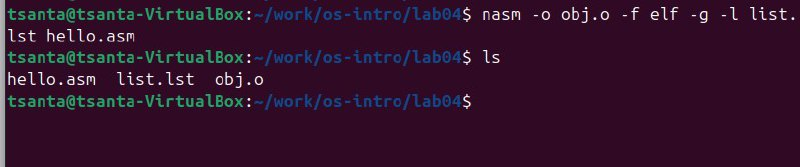


Рис. 4: рис4

##4 Компоновщиком LD

Я передам объектный файл hello.o для обработки компоновщиком LD для создания исполняемого файла hello. Далее я использую команду ls, чтобы проверить правильность выполнения команды(рис 5).

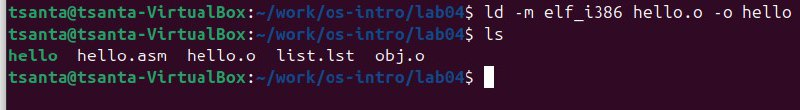


Рис. 5: рис 5

Теперь я создам файл с именем «main» с помощью данной команды.Объектный файл, из которого собран этот исполняемый файл, имеет имя obj.o(рис 6)

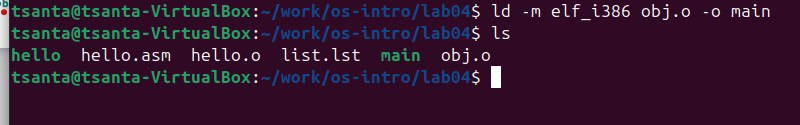


Рис. 6: рис 6

##5 Запуск исполняемого файла

Я запущу созданный исполняемый файл hello(рис 7)

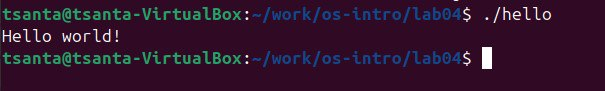


Рис. 7: рис 7

##6 Выполнение заданий для самостоятельной работы.

С помощью команды cp я создам копию файла hello.asm в текущем каталоге с именем lab4.asm(рис 8).

Рис. 8: рис 8

Рис. 8: рис 8

C помощью текстового редактора я открою файл lab4.asm и внесу изменения в программу, чтобы она отображала мое имя и фамилию(рис 9)



Рис. 9: рис 9

Текст программы я скомпилирую в объектный файл. С помощью команды ls я проверяю, что файл lab5.o создан(рис 10)

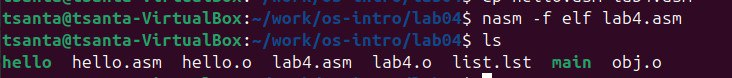


Рис. 10: рис 10

Я передам объектный файл lab5.o компоновщику LD для обработки для создания исполняемого файла lab5(рис 11).

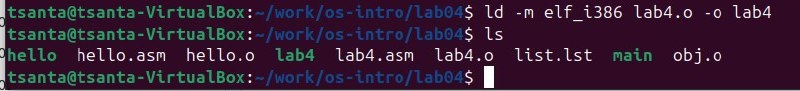


Рис. 11: рис 11

Когда я запускаю исполняемый файл lab4, мое имя и фамилия фактически отображаются на экране(рис 13)

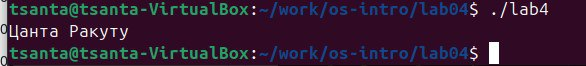


Рис. 12: рис 12

Я скопирую файлы hello.asm и lab4.asm в свой локальный репозиторий в каталоге ~/work/study/2023-2024/“Архитектура компьютера”/arch-pc/labs/lab04/report. Я сделаю это с помощью команды cd(рис 13)

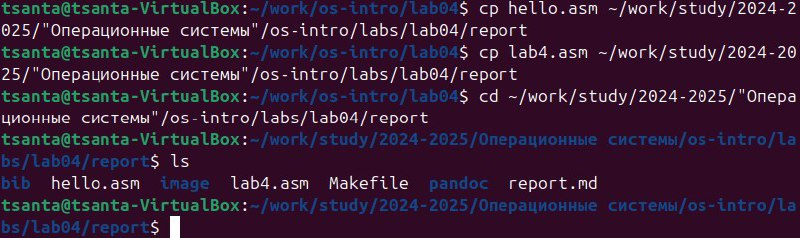


Рис. 13: рис 13

Используя команду git add и git commit, я добавлю файлы в github(рис 14)

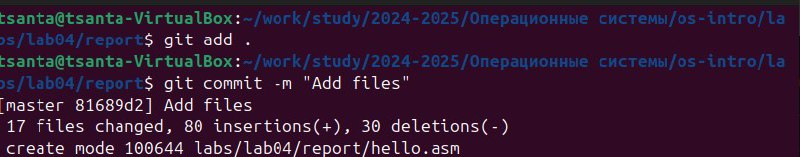


Рис. 14: рис 14

Я отправляю файлы на сервер с помощью команды git push(рис 15)

Рис. 15: рис 15

Рис. 15: рис 15

# 5 Выводы

При выполнении данной лабораторной работы я освоила процедуры компиляции и сборки программ, написанных на ассемблере NASM.

# Список литературы

Архитектура ЭВМ