Отчёта по лабораторной работе №4

Создание и процесс обработки программ на языке ассемблера NASM

Жозе Рамос Домингуш

Содержание

[1 Цель работы 1](#_Toc180877191)

[2 Задание 1](#_Toc180877192)

[3 Выполнение лабораторной работы 1](#_Toc180877193)

[3.1 Программа Hello world! 1](#_Toc180877194)

[3.2 Транаслятор NASM 2](#_Toc180877195)

[3.3 Расширенный синтаксис командной строки NASM 2](#_Toc180877196)

[3.4 Компоновщик LD 3](#_Toc180877197)

[3.5 Запуск исполняемого файла 3](#_Toc180877198)

[3.6 Задание для самостоятельной работы 4](#_Toc180877199)

[4 Выводы 5](#_Toc180877200)

# 1 Цель работы

Освоить процедуры компиляции и сборки программ, познакомиться с языком ассемблера NASM.

# 2 Задание

Написать 2 программы(Hello world, lab4(Имя Фамилия))

# 3 Выполнение лабораторной работы

## 3.1 Программа Hello world!

Создаем каталог для работы с программами на языке ассемблера NASM (рис. [1](#fig:001)).

Figure 1: Создаем каталоги с помощью команды mkdir

Figure 1: Создаем каталоги с помощью команды mkdir

Переходим в созданный каталог (рис. [2](#fig:002)).



Figure 2: Переходим в каталог с помощью команды сd

Создаем текстовый файл (рис. [3](#fig:003)).



Figure 3: Создаем текстовый файл hello.asm

Открываем данный файл в текстовом редакторе (рис. [4](#fig:004)).

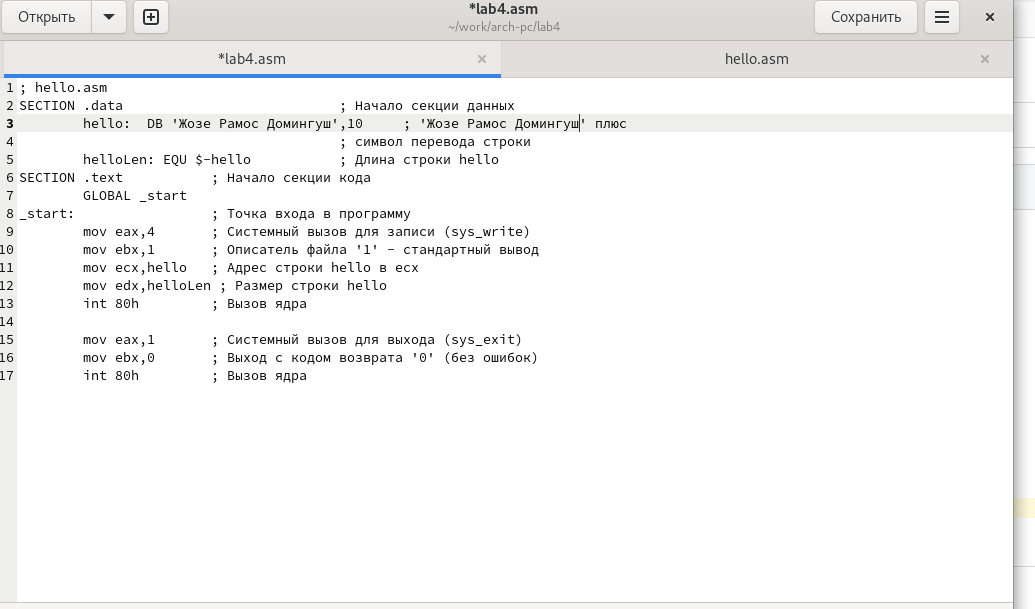


Figure 4: Открываем файл и заполняем его по примеру

## 3.2 Транаслятор NASM

Преобразуем текст программы в объектный код (рис. [5](#fig:005)).

Figure 5: Используем команду nasm

Figure 5: Используем команду nasm

Проверяем создался ли объектный файл с помощью команды ls (рис. [6](#fig:006)).

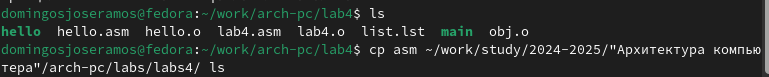


Figure 6: Проверяем работу команды

## 3.3 Расширенный синтаксис командной строки NASM

Компилируем исходный файл (рис. [7](#fig:007)).

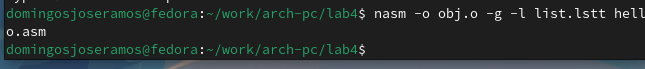


Figure 7: Преобразуем файл hello.asm в obj.o

Проверяем, как сработала команда (рис. [8](#fig:008)).

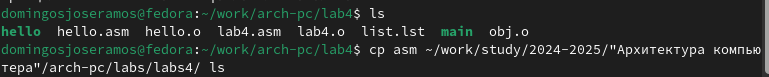


Figure 8: Проверяем создание файла командой ls

## 3.4 Компоновщик LD

Передаем объектный файл на обработку компоновщику (рис. [9](#fig:009)).



Figure 9: Используем команду ld

Проверяем создался ли исполняемый файл hello (рис. [10](#fig:010)).

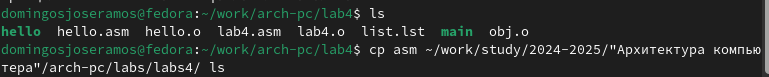


Figure 10: Используем команду ls

Передаем объектный файл на обработку компоновщику (рис. [11](#fig:011)).



Figure 11: Используем команду ld, создавая файл main

Проверяем создался ли исполняемый файл hello (рис. [12](#fig:012)).

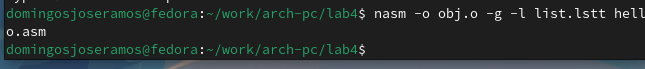


Figure 12: Используем команду ls

## 3.5 Запуск исполняемого файла

Запускаем на выполнение созданный исполняемый файл (рис. [13](#fig:013)).



Figure 13: Используем команду ./hello

## 3.6 Задание для самостоятельной работы

Создаем копию файла hello.asm (рис. [14](#fig:014)).

Figure 14: Используем команду cp

Figure 14: Используем команду cp

Открываем файл и редактируем его (рис. [15](#fig:015)).

Figure 15: Открываем файл в текстовом редакторе

Figure 15: Открываем файл в текстовом редакторе

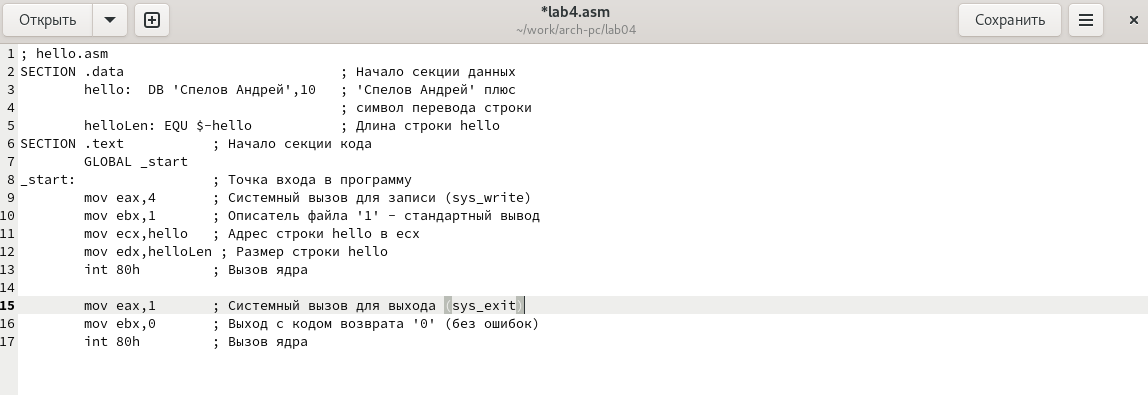


Figure 16: Редактируем файл для своего имени и фамилии

Прописывем те же команды, что и с первой программой (рис. [17](#fig:017)).



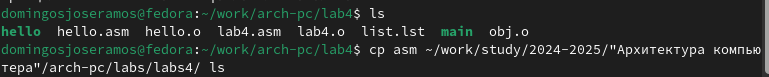


Figure 17: Прописываем команды для работы файла и запускаем программу

Копируем файлы в локальный репозиторий (рис. [18](#fig:018)).

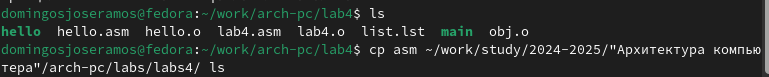


Figure 18: Копируем файлы в каталог с ЛР4

Переходим в каталог лабораторных работ и загружаем файлы на Github (рис. [19](#fig:019)).



Figure 19: Загружаем файлы

# 4 Выводы

Мы познакомились с языком ассемблера NASM и создали две работающих программы.