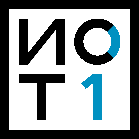
Министерство цифрового развития, связи и массовых коммуникаций РФ

Уральский технический институт связи и информатики (филиал)   
ФГБОУ ВО "Сибирский государственный университет телекоммуникаций и информатики" в г. Екатеринбурге (УрТИСИСибГУТИ)

Изображение выглядит как текст

Автоматически созданное описание

Кафедра информационных

систем и

технологий

**ОТЧЕТ**

По дисциплине «архитектура аппаратных средств»

Практическая работа № 2

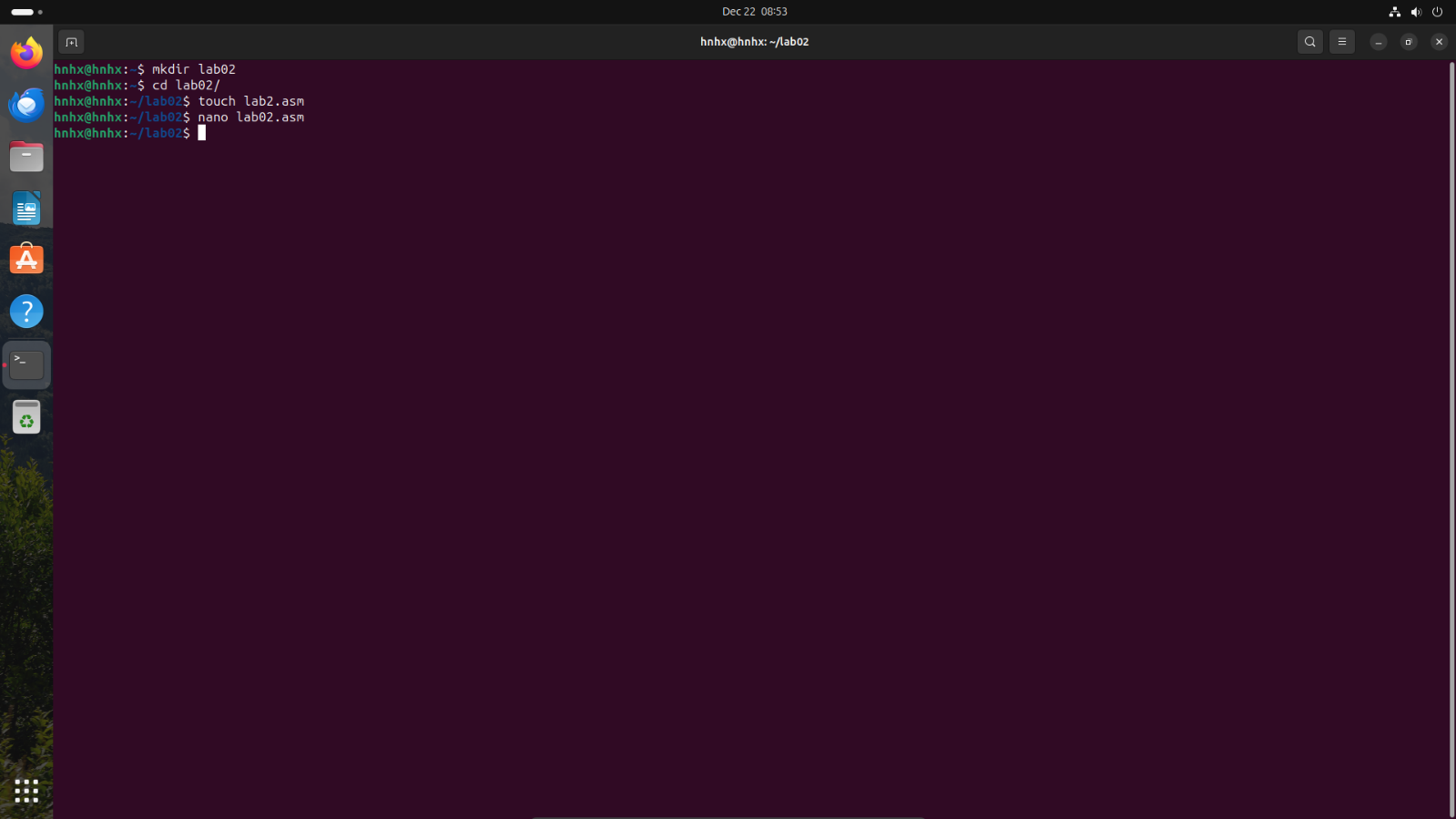
|  |  |
| --- | --- |
| Выполнил: | студент гр. 284  Шестаков И.В. |
| Проверил: | Преподаватель  Зацепин В.А. |

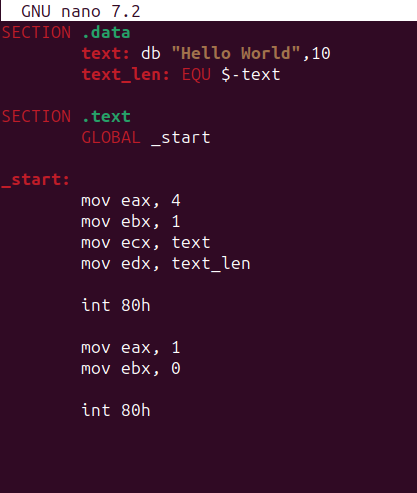
Екатеринбург, 2023

Цель: Освоение компиляции и сборки программ, написанных на  
ассемблере NASM.

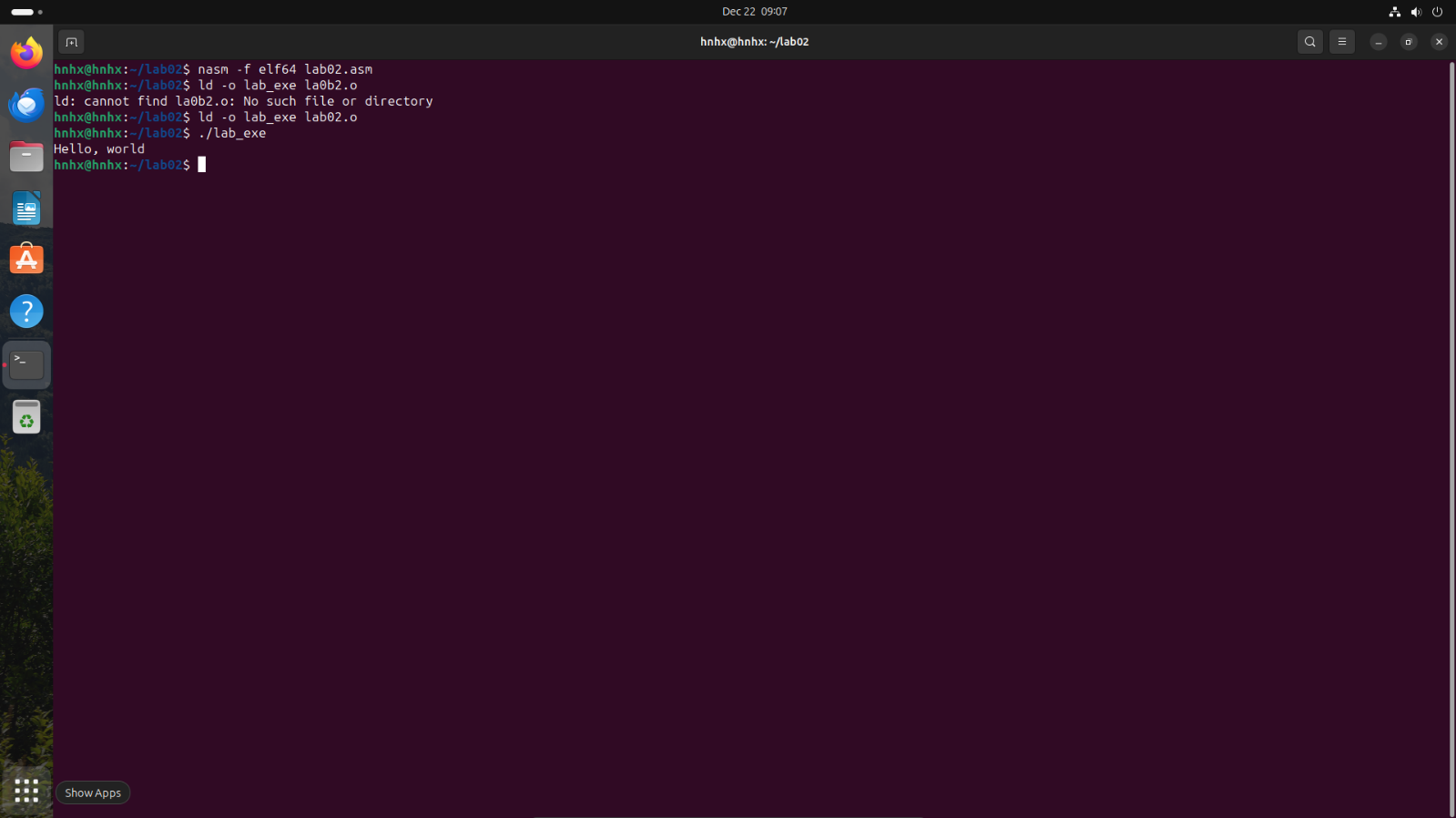
Ход работы:

1. Создал в своём домашнем каталоге новый подкаталог с именем lab02 и файл lab2.asm:  
   С помощью редактора nano ввел в файл lab2.asm текст программы hello.asm из п. 2.2.2.1

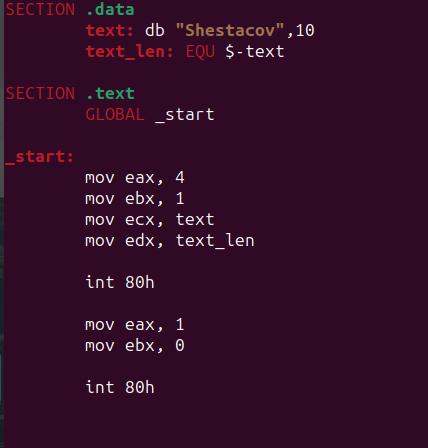


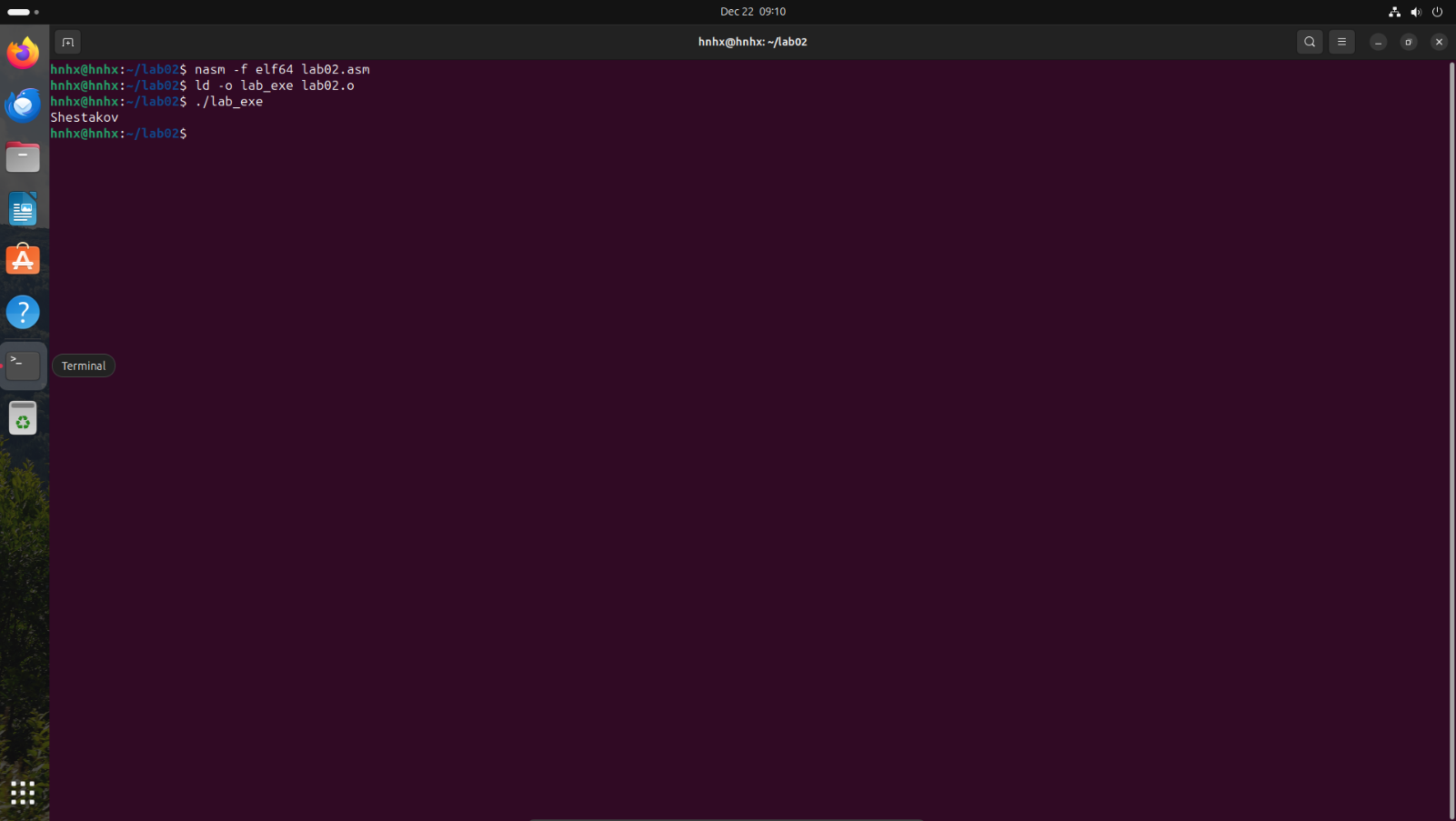


1. Оттранслировал полученный текст программы в объектный файл
2. Выполнил компоновку объектного файла и запустил получившийся  
   исполняемый файл



1. Изменил в тексте программы выводимую на экран строку с Hello world на свою фамилию, повторил пункты 2 и 3





Выводы: выполнив эту работу, я научился компилировать и собирать проекты на основе ассемблера NASM.

Ответы на контрольные вопросы:

1. Какие основные отличия ассемблерных программ от программ на языках  
   высокого уровня?

Главное отличие заключается в том, что в ассемблере идет работа с железом «напрямую» с помощью мнемоники –инструкций к процессору

Также здесь каждая команда должна располагаться на отдельной строке

1. В чём состоит отличие инструкции от директивы на языке ассемблера?

Инструкция компилируется в машинный код, а диррективы управляют работой транслятора(используются для определения данных)

1. Перечислите основные правила оформления программ на языке ассемблера

Каждая команда – на отдельной строке

Стоит обращать внимание, что значение команд зависит от регистра(SECTION != section)

Код разбивается на блоки(секции): .data ; .bss ; .text

В секции .text должен располагаться исполняемый код, а сама программа начинает работу с \_start

1. Каковы этапы получения исполняемого файла?
   1. Написать код
   2. Трансляция в объектный файл
   3. Компоновка в исполняемый файл
2. Каково назначение этапа трансляции?

Представление кода ассемблера в виде байт-кода, «понятного» для процессора

1. Каково назначение этапа компоновки?

Представление объектного кода в виде исполняемого файла, «понятного» для ОС

1. Какие файлы могут создаваться при трансляции программы, какие из  
   них создаются по умолчанию?

Могут создаваться объектный файл и листинг

По умолчанию создается объектный файл

1. каковы форматы файлов для nasm и ld?

Nasm - .o(объектный файл) ; .lst(листинг)

Ld – исполняемый файл(в linux без расширения)