МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ БЕЛАРУСЬ

БЕЛОРУССКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

## Факультет информационных технологий и робототехники

Кафедра программного обеспечения информационных систем и технологий

Отчет по лабораторной работе № 6

по дисциплине:” Системное программирование”

на тему: «Архивация, сжатие и резервное копирование в Linux»

Вариант 9

Выполнили**:**  студенты группы 10701222Махнач М. В.

Дяденькина Т.М.

Шкробот А. А.

Приняла**:** ст. пр. Давыденко Н.В.

Минск 2024

**Цель работы**: Закрепить на практике основы администрирования системы Linux, изучить атрибуты файлов и права доступа к ним, освоить работу с файлами и каталогами.

**Задание**

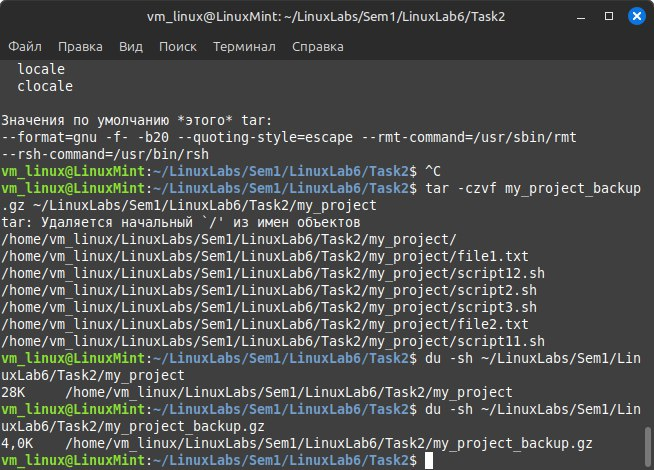
Задание 2

Проведите упаквку и сжатие вашего ранее созданного программного проекта. Перенесите сжатую версию проекта в другое пространство имен и распакуйте проект. Сравните размеры полученных файлов, а также сравните резултаты с результатами предыдущих заданий. При выполнении задания можно использовать флеш-накопитель, но предпочтительным является передача файлов между системами по сети. Сделайте вывод на основе анализа полученных результатов.

**Выполнение**

Шаг 1: Упаковка и сжатие проекта

Перейти в директорию проекта и создать сжатый архив с помощью команды tar с использованием сжатия gzip:



Здесь:

-c — для создания архива.

-z — для сжатия с помощью алгоритма gzip.

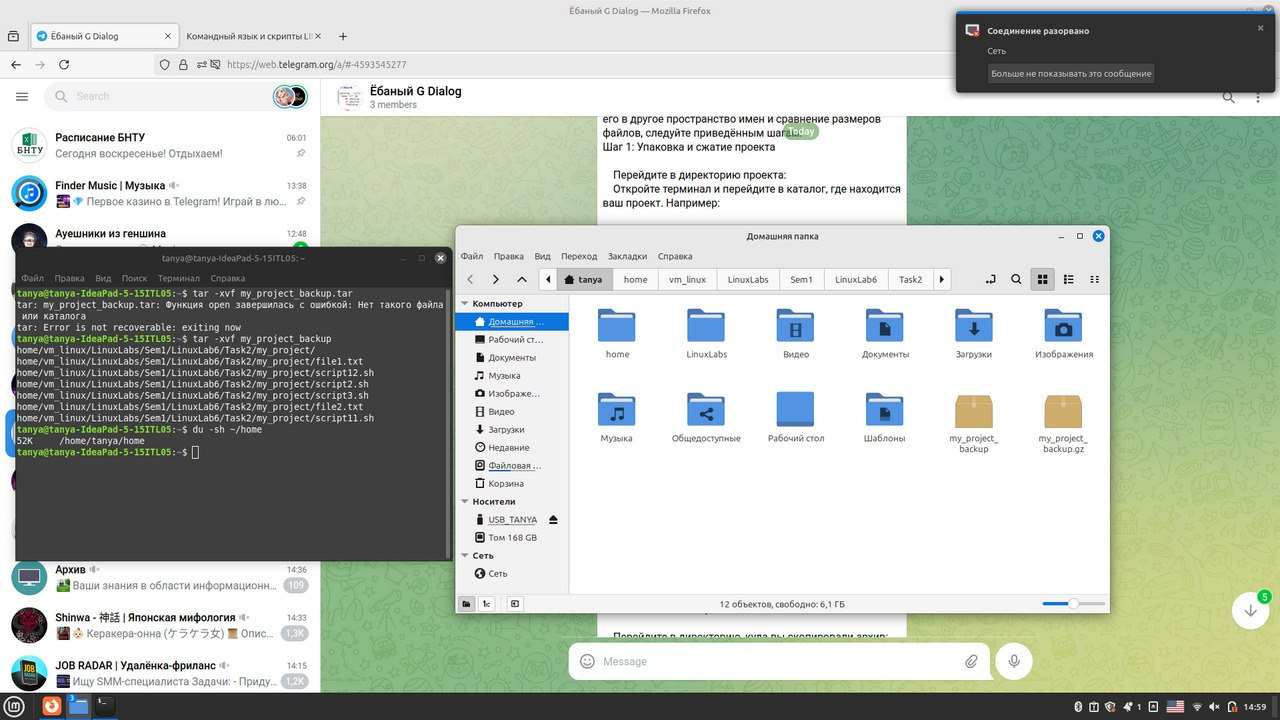
-v — для вывода информации о процессе.

-f — для указания имени выходного файла.

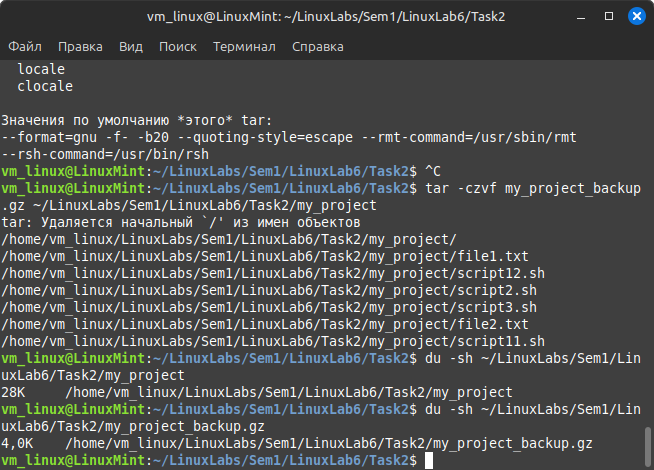
Шаг 2: Передача архива в другое пространство имен

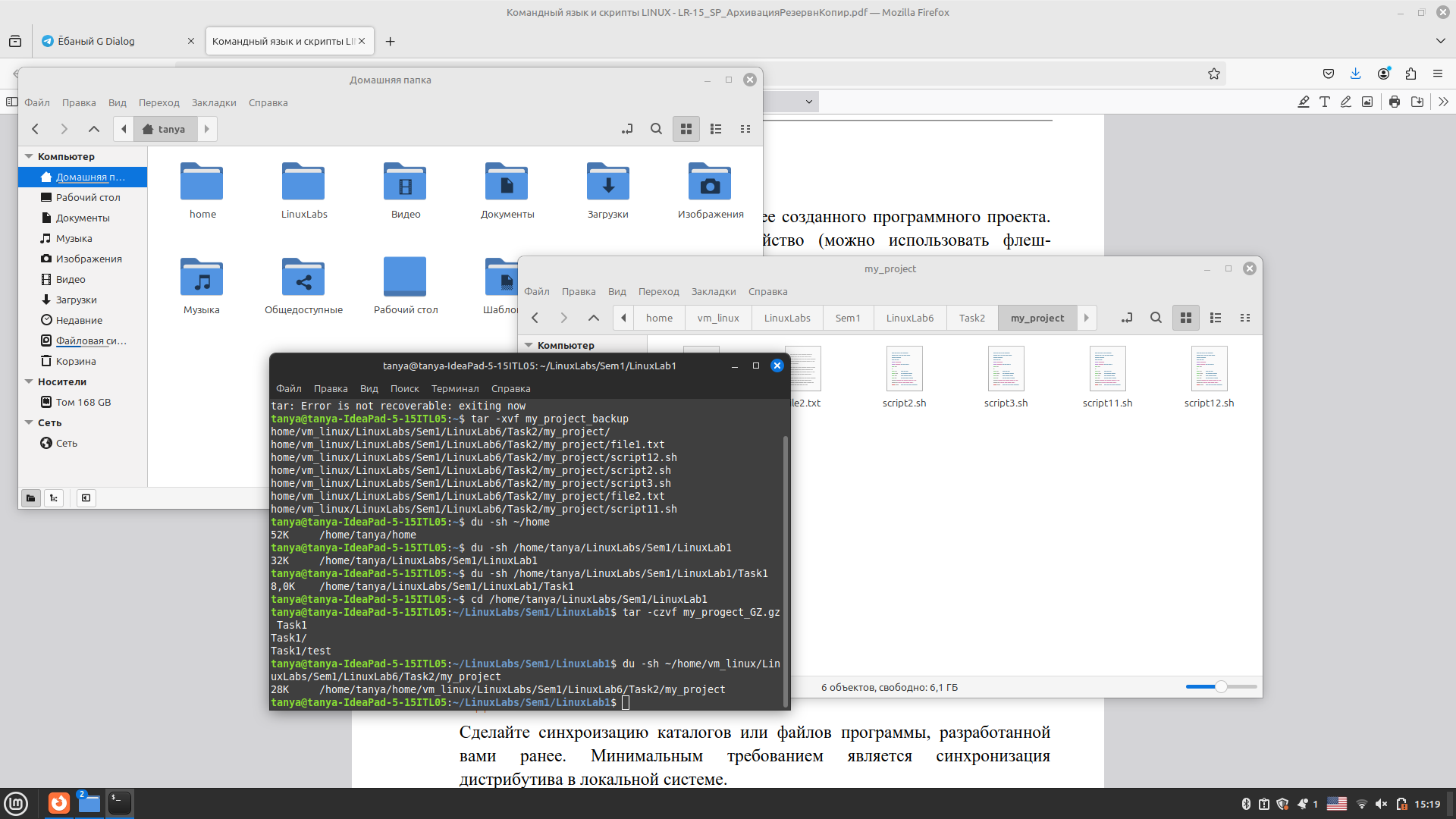
Для передачи архива в новое пространство имён использовался флэш-диск.

Шаг 3: Распаковка проекта на новом месте



Шаг 4: Сравнение размеров файлов





|  |  |
| --- | --- |
| Размер оригинальной папки проекта. | 28 Кб |
| Размер созданного сжатого архива. | 4 Кб |
| Размер распакованного проекта в новом месте | 28 Кб |

Шаг 5: Анализ и выводы

Размер сжатого файла значительно меньше, чем размер оригинала. Это подтверждает, что сжатие эффективно для уменьшения объема данных.

Размер распакованного файла соответствует размеру исходного проекта.

**Контрольные вопросы**

1. **Что такое абсолютный и относительный путь?**

Относительный и абсолютный путь — это два способа указания местоположения файла или директории в файловой системе.  
  
**Абсолютный путь** указывает полное местоположение файла или директории, начиная с корневого каталога файловой системы. Он всегда начинается с символа, представляющего корень (например, / в Linux или macOS).  
  
**Пример:**  
- /home/user/documents/file.txt — это абсолютный путь, который указывает на файл file.txt, находящийся в папке documents, которая находится в папке user, которая, в свою очередь, находится в корневом каталоге.  
  
**Относительный путь** указывает местоположение файла или директории относительно текущего рабочего каталога (директории, в которой вы в данный момент находитесь в терминале или командной строке). Он не начинается с корневого каталога.  
  
**Пример:**  
- Если ваш текущий рабочий каталог — /home/user, то относительный путь к файлу file.txt, находящемуся в директории documents, будет выглядеть как documents/file.txt.  
  
 Основные отличия:  
- **Абсолютный путь** указывает полное местоположение независимо от текущего местоположения, тогда как **относительный путь** учитывает ваше текущее местоположение.  
- Абсолютные пути обычно менее удобны для быстрого навигации, но более стабильны, так как не зависят от контекста текущего каталога.