# **Домашнее задание**

|  |  |
| --- | --- |
| Дисциплина | Семинар наставника |
| Тема | Тема 4. Работа с оболочкой Bash для решения задач в области инженерии данных |
| Форма проверки | **Задание с индивидуальной проверкой преподавателем.**  После дедлайна преподаватель проверит задания в течении полутора недель.  *Совет: выполняйте домашнее задание сразу после изучения темы* |
| Имя преподавателя | Владислав Шевченко |
| Время выполнения | 2 часа |
| Цель задания | Закрепить навык работы с Bash |
| Инструменты для выполнения ДЗ | Ссылка на репозиторий на GitHub. Настройки репозитория должны быть public. Дополнительно можно приложить скриншоты. |
| Правила приёма работы | Отправьте ссылку на репозиторий в LMS или прикрепите файл со скриншотами выполнения задания. |
| Критерии оценки | За успешное выполнение 7 задач вы получите 8 баллов. За выполнение меньшего количества задач баллы будут уменьшаться пропорционально. |
| Дедлайн | *1 ноября 2025* |

## Задачи

#### 1. Функционал Bash

* **Задание**: Напишите Bash-скрипт, который выполняет следующие действия:
  1. Создаёт список всех файлов в текущей директории, указывая их тип (файл, каталог и т.д.).
  2. Проверяет наличие определённого файла, переданного как аргумент скрипта, и выводит сообщение о его наличии или отсутствии.
  3. Использует цикл for для вывода информации о каждом файле: его имя и права доступа.

#### 2. Переменная PATH

* **Задание**:
  1. Напишите скрипт, который выводит текущее значение переменной PATH и добавляет в неё новую директорию, переданную в качестве аргумента.
  2. Объясните, почему изменения переменной PATH, сделанные через терминал, временные, и предложите способ сделать их постоянными. Добавьте команду в файл .bashrc и продемонстрируйте, как перезапустить терминал для применения изменений.

#### 3. Управляющие конструкции (условия и циклы)

* **Задание**: Напишите скрипт, который запрашивает у пользователя ввод числа и затем:
  + Использует if, чтобы проверить, является ли число положительным, отрицательным или нулем, и выводит соответствующее сообщение.
  + Использует while для подсчёта от 1 до введенного числа (если оно положительное).

#### 4. Работа с функциями

* **Задание**: Создайте скрипт с функцией, которая принимает в качестве аргумента строку и выводит её с префиксом "Hello, ". Напишите ещё одну функцию, которая принимает два числа и возвращает их сумму. Вызовите обе функции в скрипте и продемонстрируйте результат.

#### 5. Управление процессами и фоновое выполнение

* **Задание**: Напишите скрипт, который запускает три команды sleep с разными временами в фоновом режиме. Используйте команды jobs, fg, bg, чтобы продемонстрировать управление этими задачами. Опишите, что вы наблюдали.

#### 6. Ввод/вывод и перенаправление

* **Задание**: Создайте скрипт, который выполняет следующие действия:
  1. Читает данные из файла input.txt.
  2. Перенаправляет вывод команды wc -l (подсчет строк) в файл output.txt.
  3. Перенаправляет ошибки выполнения команды ls для несуществующего файла в файл error.log.

#### 7. Использование alias и автодополнение

* **Задание**: Создайте alias для команды ls -la и назовите его ll. Напишите команду, чтобы сделать alias постоянным, и объясните, где она должна быть добавлена. Продемонстрируйте использование автодополнения на примере команды cd.

**Дополнительные сложные задачи   
(для продвинутых студентов)**

1. **Менеджер резервного копирования**: Создайте скрипт, который:
   * Создаёт резервную копию всех файлов в указанной директории, добавляя к именам файлов текущую дату.
   * Поддерживает сохранение логов операций в отдельный файл и уведомляет пользователя об успешном завершении с указанием количества файлов.
2. **Мониторинг системы**: Напишите скрипт для мониторинга ресурсов системы:
   * Сбор данных о загрузке процессора, использовании памяти и дискового пространства.
   * Уведомление пользователя, если использование памяти превышает 80%, с выводом текущих процессов, потребляющих наибольшее количество ресурсов.
3. **Автоматическая сортировка файлов**: Напишите скрипт, который автоматически сортирует файлы в указанной директории:
   * Файлы с расширениями .jpg, .png и .gif перемещает в папку Images.
   * Документы (.txt, .pdf, .docx) — в папку Documents.
   * Скрипт должен запускаться автоматически через cron каждую ночь и вести лог всех перемещений.