Отчёт по лабораторной работе №13

*Дисциплина: Операционные системы*

Долгаев Евгений Сергеевич

Содержание

Список иллюстраций

Список таблиц

# 1 Цель работы

Изучить основы программирования в оболочке ОС UNIX. Научится писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.

# 2 Задание

1. Используя команды getopts grep, написать командный файл, который анализирует командную строку с ключами:

* -i inputfile — прочитать данные из указанного файла;
* -o outputfile — вывести данные в указанный файл;
* -p шаблон — указать шаблон для поиска;
* -C — различать большие и малые буквы;
* -n — выдавать номера строк. а затем ищет в указанном файле нужные строки, определяемые ключом -p.

1. Написать на языке Си программу, которая вводит число и определяет, является ли оно больше нуля, меньше нуля или равно нулю. Затем программа завершается с помощью функции exit(n), передавая информацию в о коде завершения в оболочку. Командный файл должен вызывать эту программу и, проанализировав с помощью команды $?, выдать сообщение о том, какое число было введено.
2. Написать командный файл, создающий указанное число файлов, пронумерованных последовательно от 1 до 𝑁 (например 1.tmp, 2.tmp, 3.tmp,4.tmp и т.д.). Число файлов, которые необходимо создать, передаётся в аргументы командной строки. Этот же командный файл должен уметь удалять все созданные им файлы (если они существуют).
3. Написать командный файл, который с помощью команды tar запаковывает в архив все файлы в указанной директории. Модифицировать его так, чтобы запаковывались только те файлы, которые были изменены менее недели тому назад (использовать команду find).

# 3 Выполнение лабораторной работы

Далее в отчете будут прикреплены скриншоты выполнения работы.

## 3.1 Задача 1

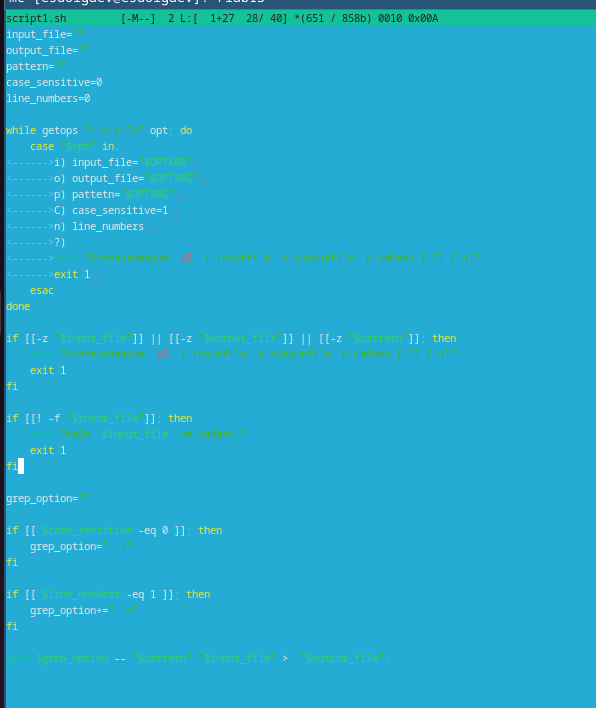


Рис. 1: Задача 1. Код

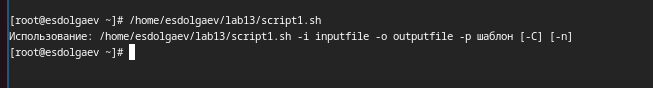


Рис. 2: Задача 1. Результат выполнения

## 3.2 Задача 2

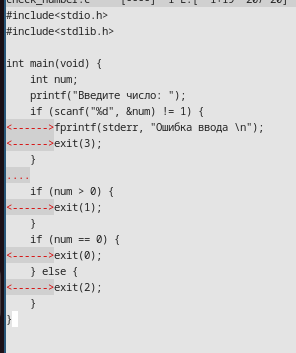


Рис. 3: Задача 2. Код на C

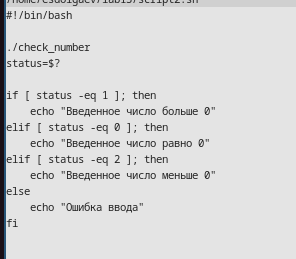


Рис. 4: Задача 2. Код на bash

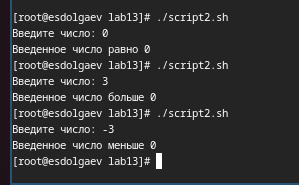


Рис. 5: Задача 2. Результат выполнения

## 3.3 Задача 3

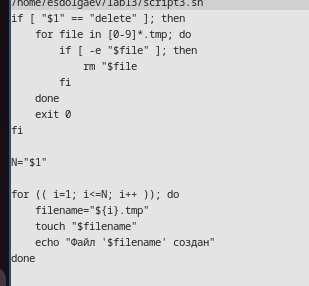


Рис. 6: Задача 3. Код

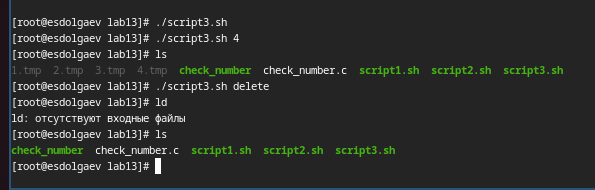


Рис. 7: Задача 3. Результат выполнения

## 3.4 Задача 4

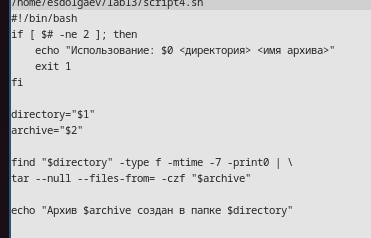


Рис. 8: Задача 4. Код

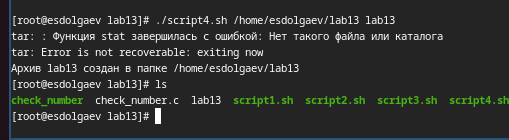


Рис. 9: Задача 4. Результат выполнения

# 4 Выводы

Таким образом, я научился писать более сложные программы с использованием логических управляющих конструкций и циклов.