

Презентация к лабораторной работе №13

Операционные системы

Бережной И. А., НКАбд-01-23

Российский университет дружбы народов, Москва, Россия

Для программиста необходимо уметь писать рабочие скрипты, чтобы оптимизировать процесс работы

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

1. Написать командный файл, ищущий нужные строки;
2. Написать программу на языке Си, определяющую, какое число было введено, и командный файл, вызывающий эту программу;
3. Написать командный файл, создающий указанное количество файлов, пронумерованных последовательно от 1 до N;
4. Написать командный файл, который сжимаем указанные файлы в архив.

Командный процессор (командная оболочка, интерпретатор команд shell) — это программа, позволяющая пользователю взаимодействовать с операционной системой компьютера. В операционных системах типа UNIX/Linux наиболее часто используются следующие реализации командных оболочек:

- оболочка Борна (Bourne shell или sh) — стандартная командная оболочка UNIX/Linux, содержащая базовый, но при этом полный набор функций;

- С-оболочка (или `cs`h) — надстройка на оболочке Борна, использующая С-подобный синтаксис команд с возможностью сохранения истории выполнения команд;
- оболочка Корна (или `k`sh) — напоминает оболочку С, но операторы управления программой совместимы с операторами оболочки Борна;
- BASH — сокращение от Bourne Again Shell (опять оболочка Борна), в основе своей совмещает свойства оболочек С и Корна (разработка компании Free Software Foundation).

Создадим файл в терминале и присвоим ему право на исполнение. Напишем скрипт, который анализирует командную строку с ключами и ищет в указанном файле нужные строки, определённые после ключа **-p**. Создадим текстовый файл, в котором будем искать строки. Запустим скрипт и получим вот такой результат.

Создадим второй командный файл и файл с расширением **.c**. В последнем напишем команду, которая будет проверять, является ли число положительным, отрицательным или нулём. Напишем скрипт, который будет вызывать этот файл и выводить в терминал результат в виде текста. Проверим работу файлов.

Теперь напишем командный файл, создающий указанное число файлов, пронумерованных последовательно от 1 до N, при этом количество файлов будет передаваться в аргументы командной строки. Проверим, работает ли скрипт.

И, наконец, напишем командный файл, который с помощью команды `tar` запаковывает в архив все файлы в указанной директории. Запустим скрипт через командную строку.

Мы изучили основы программирования в оболочке ОС UNIX и научились писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.