Лабораторная работа № 11

Программирование в командном процессоре ОС UNIX. Командные файлы

Маметкадыров Ынтымак

# Цель работы

Изучить основы программирования в оболочке ОС UNIX/Linux. Научиться писать небольшие командные файлы.

# Выполнение лабораторной работы

1. Создали директорию backup в нашем домашнем каталоге (рис. 1).
2. Создали файл backup.sh (рис. 1).

Рис. 1. Создание файл backup.sh

Рис. 1. Создание файл backup.sh

1. Написали скрипт (рис. 2), который при запуске будет делать резервную копию самого себя в директорию backup в нашем домашнем каталоге (рис. 3).

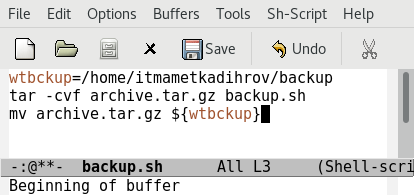


Рис. 2. Содержимое файл backup.sh

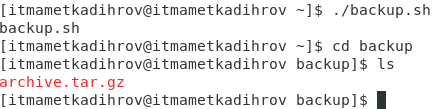


Рис. 3. Созданный архив

1. Создали командный файл print.sh (рис. 4), который последовательно распечатывает значения всех переданных аргументов, в том числе превышающих десять (рис. 5).

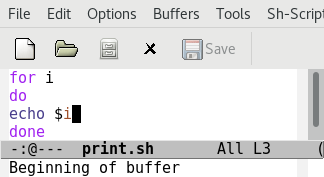


Рис. 4. Командный файл print.sh

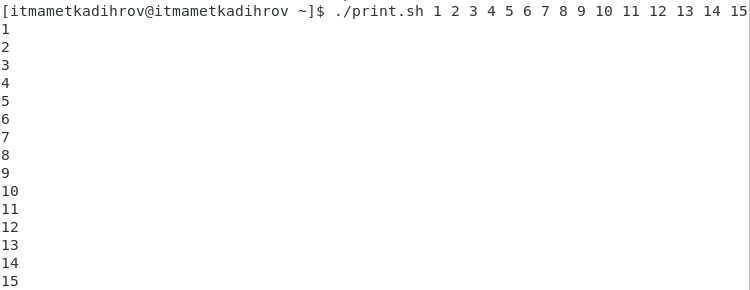


Рис. 5. Работа командного файла print.sh

1. Создали командный файл ls.sh - аналог команды ls (рис. 6). Этот файл выдает информацию о нужном каталоге и выводит информацию о возможностях доступа к файлам этого каталога без использования команды ls и dir (рис. 7).

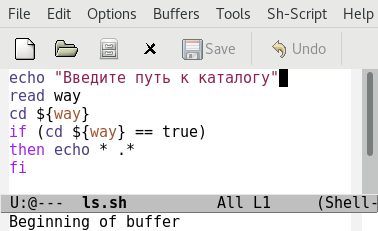


Рис. 6. Командный файл ls.sh

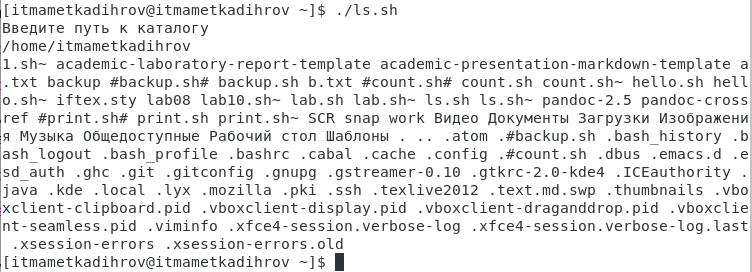


Рис. 7. Запуск командного файла ls.sh

1. Написали командный файл (рис. 8), который получает в качестве аргумента командной строки формат файла (.txt, .doc, .jpg, .pdf и т.д.) и вычисляет количество таких файлов в указанной директории. Путь к директории также передаётся в виде аргумента командной строки (рис. 9).

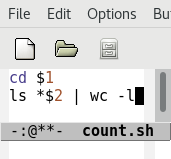


Рис. 8. Командный файл сount.sh

Рис. 9. Работа командного файла сount.sh

Рис. 9. Работа командного файла сount.sh

# Вывод

Изучили основы программирования в оболочке ОС UNIX/Linux. Научились писать небольшие командные файлы.

# Ответы на контрольные вопросы

1. 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 csh) — надстройка на оболочкой Борна, использующая Сподобный синтаксис команд с возможностью сохранения истории выполнения команд;
* – оболочка Корна (или ksh) — 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).

1. POSIX (Portable Operating System Interface for Computer Environments) — набор стандартов описания интерфейсов взаимодействия операционной системы и прикладных программ.
2. Командный процессор bash обеспечивает возможность использования переменных типа строка символов. Имена переменных могут быть выбраны пользователем. Пользователь имеет возможность присвоить переменной значение некоторой строки символов.

* Оболочка bash позволяет работать с массивами. Для создания массива используется команда set с флагом -A. За флагом следует имя переменной, а затем список значений, разделённых пробелами.

1. Оболочка bash поддерживает встроенные арифметические функции. Команда let является показателем того, что последующие аргументы представляют собой выражение, подлежащее вычислению.

* Команда read позволяет читать значения переменных со стандартного ввода.

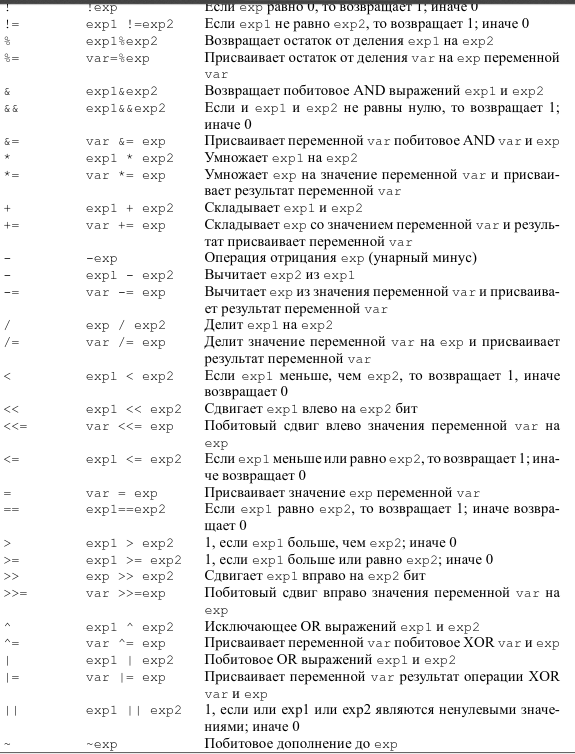
1. 

Рис. 6. Арифметические операторы оболочки bash

1. Для облегчения программирования можно записывать условия оболочки bash в двойные скобки — (( )).
2. PS1, PS2, HOME, IFS, MAIL, TERM, LOGNAME.
3. Такие символы, как ’ < > \* ? |  " &, являются метасимволами и имеют для командного процессора специальный смысл.
4. Снятие специального смысла с метасимвола называется экранированием метасимвола. Экранирование может быть осуществлено с помощью предшествующего метасимволу символа , который, в свою очередь, является метасимволом.
5. Последовательность команд может быть помещена в текстовый файл. Такой файл называется командным. Далее этот файл можно выполнить по команде:

* bash командный\_файл [аргументы]

1. Для этого существует ключевое слово function, после которого следует имя функции и список команд, заключённых в фигурные скобки.
2. test -f file — истина, если файл file существует;

* test -d file — истина, если файл file является каталогом.

1. Для создания массива используется команда set с флагом -A.

* Если использовать typeset -i для объявления и присвоения переменной, то при последующем её применении она станет целой.
* Изъять переменную из программы можно с помощью команды unset.

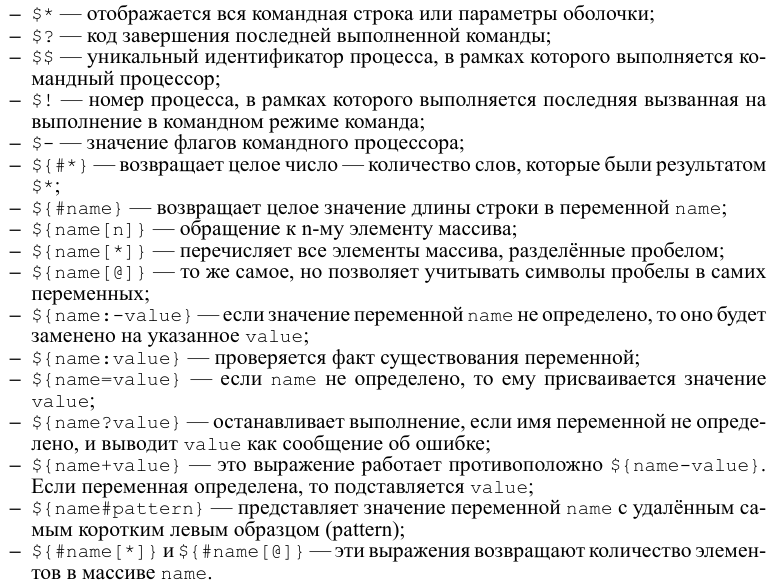
1. Чтобы передать параметры в командный файл неоходимо после названия командного файла через пробел написать передаваемые параметры.
2. 

Рис. 7. Арифметические операторы оболочки bash