Отчет по лабораторной работе №10

Дисциплина: операционные системы

Лобанова Полина Иннокентьевна

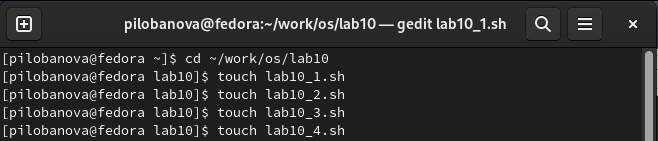
Содержание

# 1 Цель работы

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

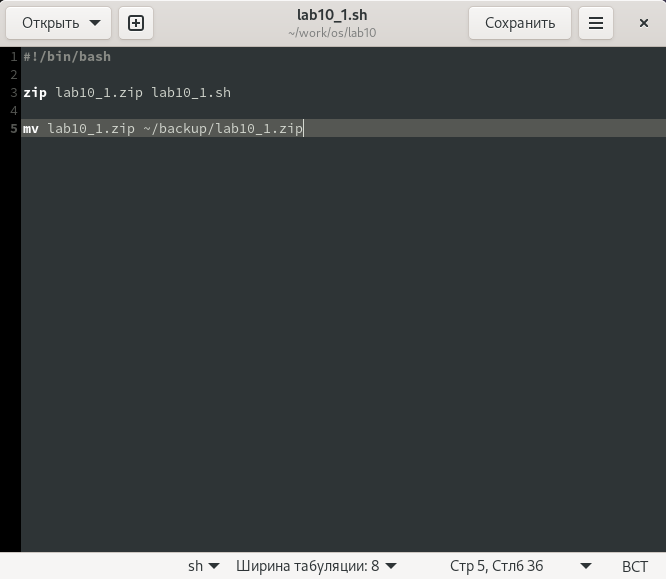
# 2 Выполнение лабораторной работы

1. Создадим необходимые файлы, чтобы в дальнейшем их заполнить.

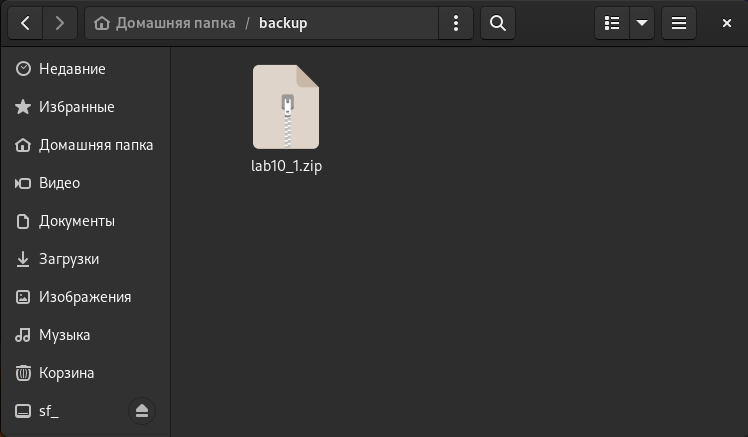


*Создание файлов.*

1. Напишем скрипт, который при запуске будет делать резервную копию самого себя в другую директорию backup в нашем домашнем каталоге. При этом файл должен архивироваться одним из архиваторов на выбор zip, bzip2 или tar.

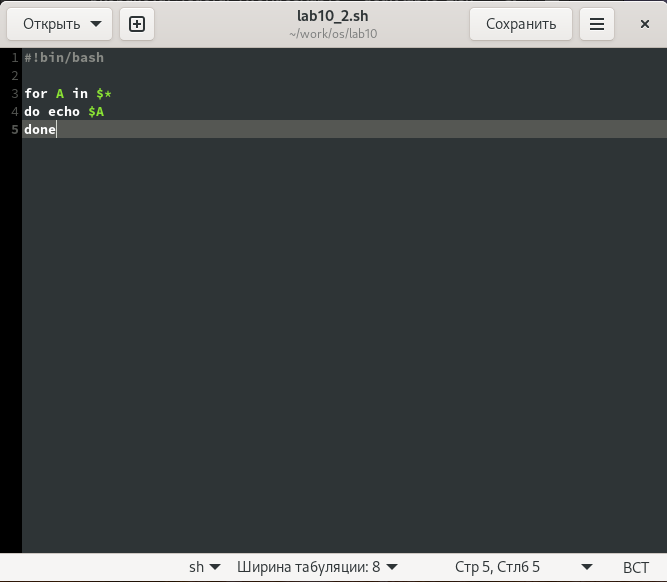


*Текст командного файла.*

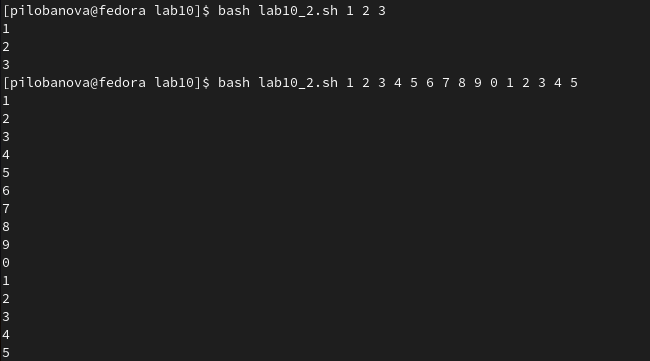


*Результат.*

1. Напишем пример командного файла, обрабатывающего любое произвольное число аргументов командной строки, в том числе превышающее десять. Например, скрипт будет последовательно распечатывать значения всех переданных аргументов.

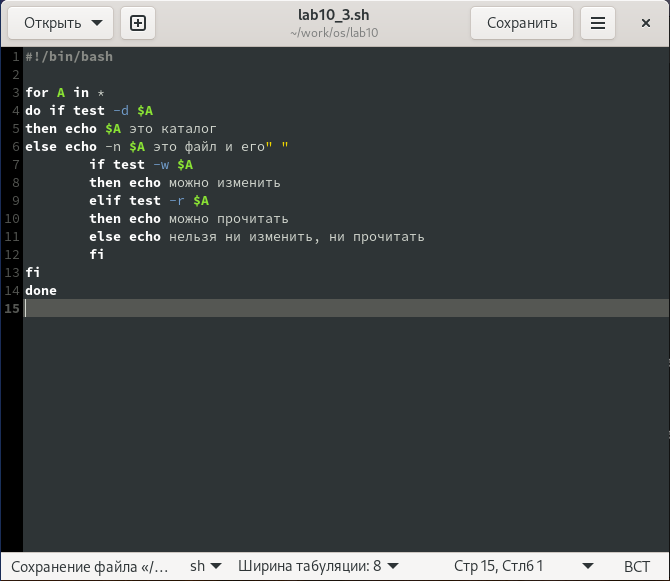


*Текст командного файла.*



*Результат.*

1. Напишем командный файл — аналог команды ls (без использования самой этой команды и команды dir) так, чтобы он выдавал информацию о нужном каталоге и выводил информацию о возможностях доступа к файлам этого каталога.

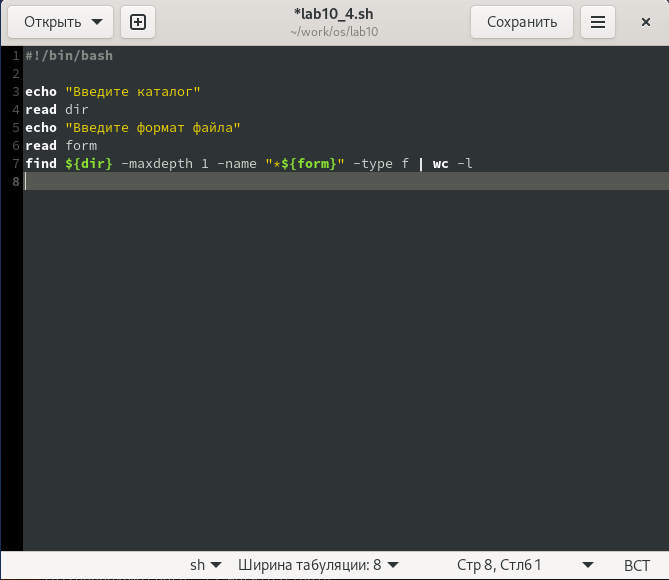


*Текст командного файла.*

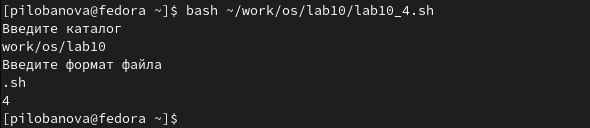


*Результат.*

1. Напишем командный файл, 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.



*Текст командного файла.*



*Результат.*

# 3 Контрольные вопросы

1. Объясните понятие командной оболочки. Приведите примеры командных оболочек. Чем они отличаются?

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 С-подобный 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. Что такое POSIX?

POSIX (Portable Operating System Interface for Computer Environments) — набор стандартов описания интерфейсов взаимодействия операционной системы и прикладных программ

1. Как определяются переменные и массивы в языке программирования bash?

mark=/usr/andy/bin Данная команда присваивает значение строки символов /usr/andy/bin переменной mark типа строкасимволов. Для создания массива используется команда set с флагом -A. За флагом следует имя переменной, а затем список значений, разделённых пробелами. Например,

set -A states Delaware Michigan “New Jersey”

1. Каково назначение операторов let и read?

Команда let является показателем того, что последующие аргументы представляют собой выражение, подлежащее вычислению. Команда read позволяет читать значения переменных со стандартного ввода

1. Какие арифметические операции можно применять в языке программирования bash?

Простейшими математическими выражениями являются сложение (+), вычитание (-), умножение (\*), целочисленное деление (/) и целочисленный остаток от деления (%).

1. Что означает операция (( ))?

Для облегчения программирования можно записывать условия оболочки bash в двойные скобки — (( )).

1. Какие стандартные имена переменных Вам известны?

Переменные PS1 и PS2 предназначены для отображения промптера командного процессора. PS1 — это промптер командного процессора, по умолчанию его значение равно символу $ или #. Если какая-то интерактивная программа, запущенная командным процессором, требует ввода, то используется промптер PS2. Он по умолчанию имеет значение символа >. Другие стандартные переменные: • HOME — имя домашнего каталога пользователя. Если команда cd вводится без аргументов, то происходит переход в каталог, указанный в этой переменной. • IFS — последовательность символов, являющихся разделителями в командной строке,например, пробел, табуляция и перевод строки (new line). • MAIL — командный процессор каждый раз перед выводом на экран промптера проверяет содержимое файла, имя которого указано в этой переменной, и если содержимое этого файла изменилось с момента последнего ввода из него, то перед тем как вывести на терминал промптер, командный процессор выводит на терминал сообщение You have mail (у Вас есть почта). • TERM — тип используемого терминала. • LOGNAME — содержит регистрационное имя пользователя, которое устанавливается автоматически при входе в систему.

1. Что такое метасимволы?

Такие символы, как ’ < > \* ? | ” &, являются метасимволами и имеют для командного процессора специальный смысл

1. Как экранировать метасимволы?

Снятие специального смысла с метасимвола называется экранированием метасимвола. Экранирование может быть осуществлено с помощью предшествующего метасимволу символа , который, в свою очередь, является метасимволом.

1. Как создавать и запускать командные файлы?

Командный файл можно создать с помощьб какого-либо редактора, затем сделать его исполняемым и запустить его из терминала, введя “./название файла”.

1. Как определяются функции в языке программирования bash?

С помощью ключевого слова function.

1. Каким образом можно выяснить, является файл каталогом или обычным файлом?

Вводим команду ls -lrt и если первым в правах доступа стоит d то это каталог. Иначе это файл

1. Каково назначение команд set, typeset и unset?

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. Как передаются параметры в командные файлы?

При вызове командного файла на выполнение параметры ему могут быть переданы точно таким же образом, как и выполняемой программе. С точки зрения командного файла эти параметры являются позиционными. Символ $ является метасимволом командного процессора. Он используется, в частности, для ссылки на параметры, точнее, для получения их значений в командном файле. В командный файл можно передать до девяти параметров. • $\* — отображается вся командная строка или параметры оболочки; • $? — код завершения последней выполненной команды; • $$ — уникальный идентификатор процесса, в рамках которого выполняется командный процессор; • $! — номер процесса, в рамках которого выполняется последняя вызванная на выполнение в командном режиме команда; • $- — значение флагов командного процессора; • ${#} — возвращает целое число — количество слов, которые были результатом $; • ${#name} — возвращает целое значение длины строки в переменной name; • ${name[n]} — обращение к n-му элементу массива; • ${name[\*]} — перечисляет все элементы массива, разделённые пробелом; • ${name[@]} — то же самое, но позволяет учитывать символы пробелы в самих переменных; • ${name:-value} — если значение переменной name не определено, то оно будет заменено на указанное value; • ${name:value} — проверяется факт существования переменной; • ${name=value} — если name не определено, то ему присваивается значение value; • ${name?value} — останавливает выполнение, если имя переменной не определено, и выводит value как сообщение об ошибке; • ${name+value} — это выражение работает противоположно ${name-value}. Если переменная определена, то подставляется value; • ${name#pattern} — представляет значение переменной name с удалённым самым коротким левым образцом (pattern); • ${#name[\*]} и ${#name[@]} — эти выражения возвращают количество элементов в массиве name.

# 4 Вывод

Я изучила основы программирования в оболочке ОС UNIX/Linux и научилась писать небольшие командные файлы.