**Орловский государственный университет имени И. С. Тургенева**

Институт приборостроения, автоматизации и информационных технологий

Кафедра информационных систем и цифровых технологий

Дисциплина «Операционные системы»

Отчет к лабораторной работе № 3

«Создание и выполнение командных файлов  
в среде ОС Linux»

Выполнила:

Студентка группы 21ПГ

Банных Мария Алексеевна

Принял:

Доцент кафедры   
информационных систем   
и цифровых технологий  
Загородних Николай Анатольевич  
  
И.о. зав. кафедрой   
информационных систем   
и цифровых технологий  
Рыженков Денис Викторович

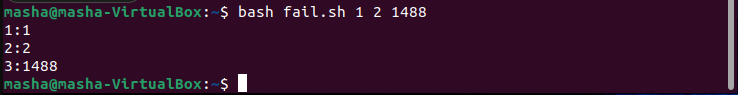
Орёл 2022 год

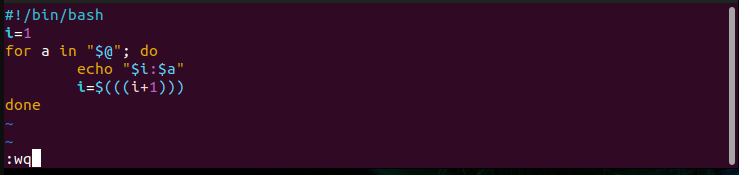
Составьте и выполните shell - программы, включающей следующие действия:

1. Вывод на экран списка параметров командной строки с указанием номера каждого параметра.

Что делала:

* В текстовом редакторе vi создала файл с программой
* С помощью оболочки bush перевели файл в машинный код и получили необходимые данные.

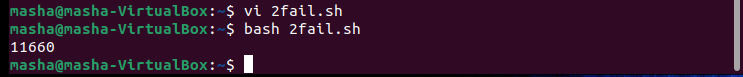


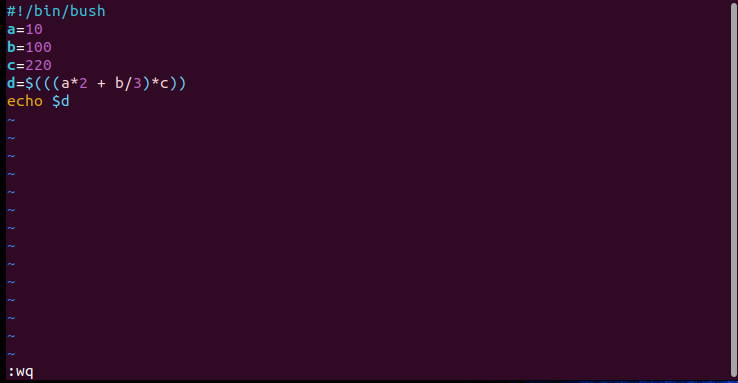


1. Присвоение переменным А, В и С значений 10, 100 и 200, вычисление и вывод результатов по формуле D=(A\*2 + B/3)\*C.

Что делала:

* В текстовом редакторе vi создала файл с программой
* С помощью оболочки bush перевели файл в машинный код и получили ответ

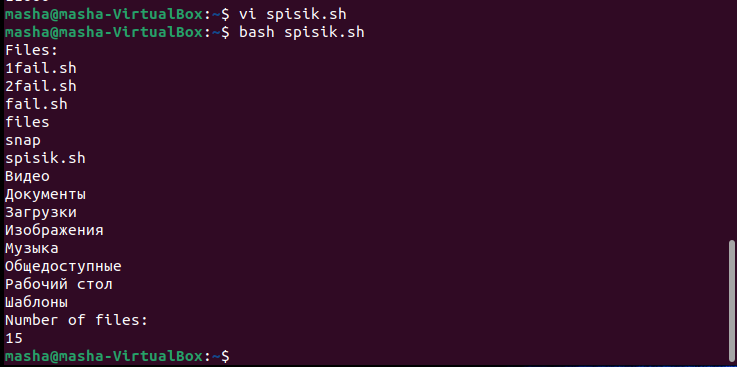


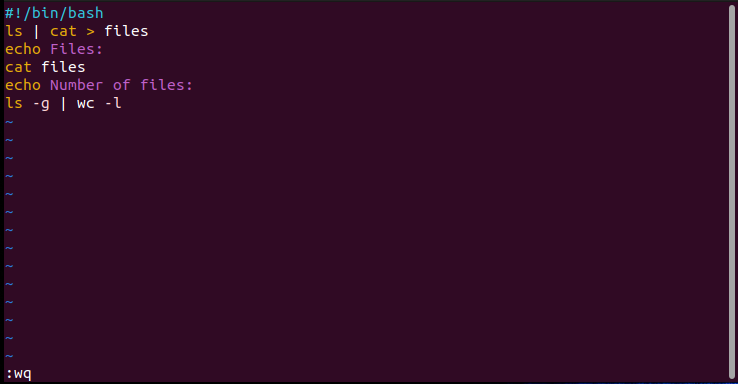


1. Формирование файла со списком файлов в домашнем каталоге, вывод на экран этого списка в алфавитном порядке и общего количества файлов.

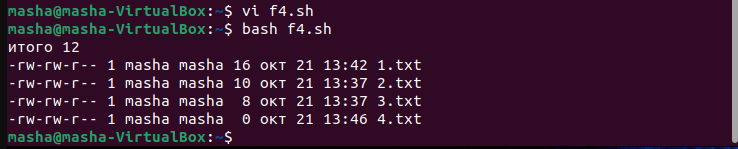
Что делала:

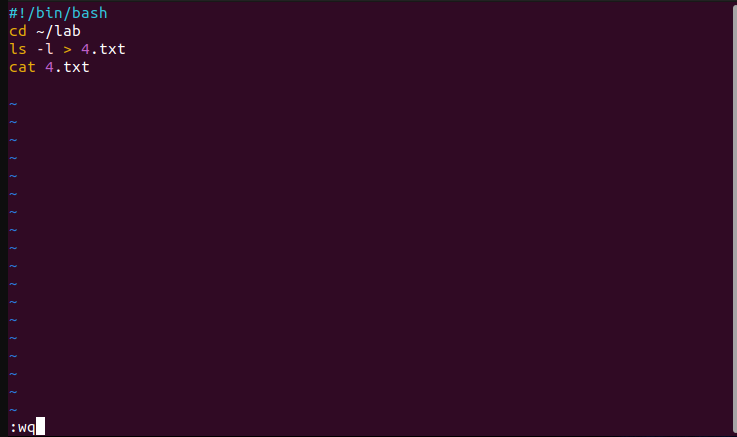
* С помощью текстового редактора vi создала файл с программой и запустила его через оболочку bush



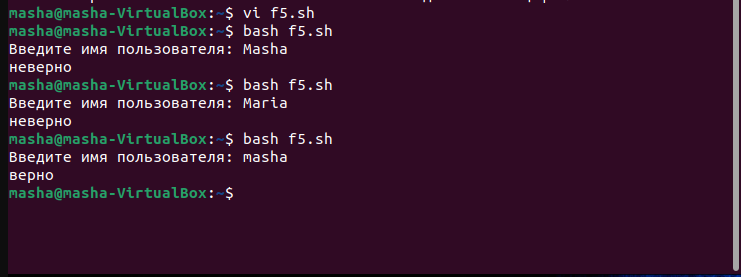


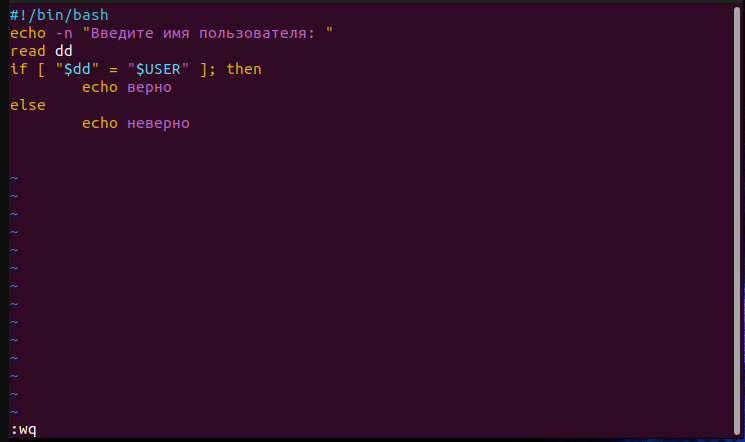
1. Переход в другой каталог, формирование файла с листингом каталога и возвращение в исходный каталог.



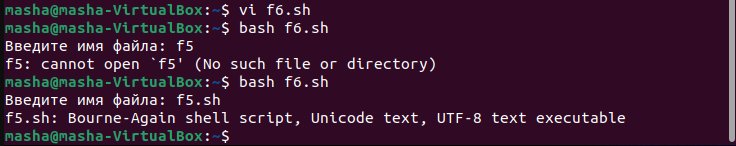


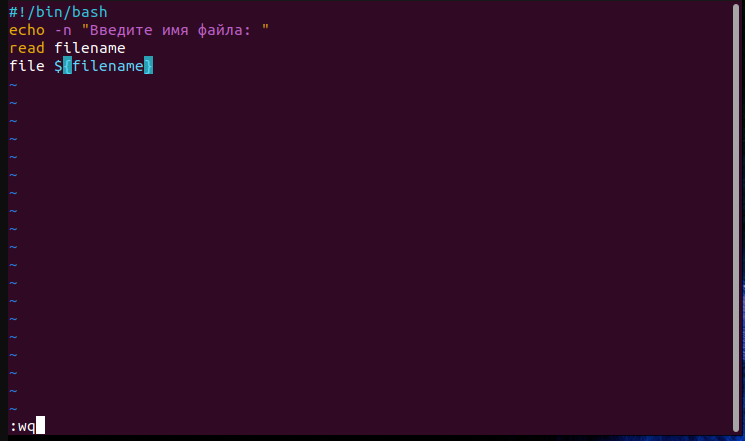
1. Запрос и ввод имени пользователя, сравнение с текущим логическим именем пользователя и вывод сообщения: верно/неверно.





1. Запрос и ввод имени файла в текущем каталоге и вывод сообщения о типе файла.

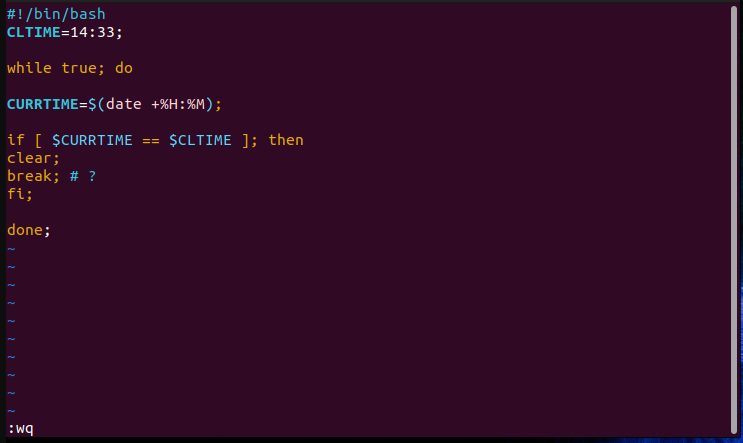




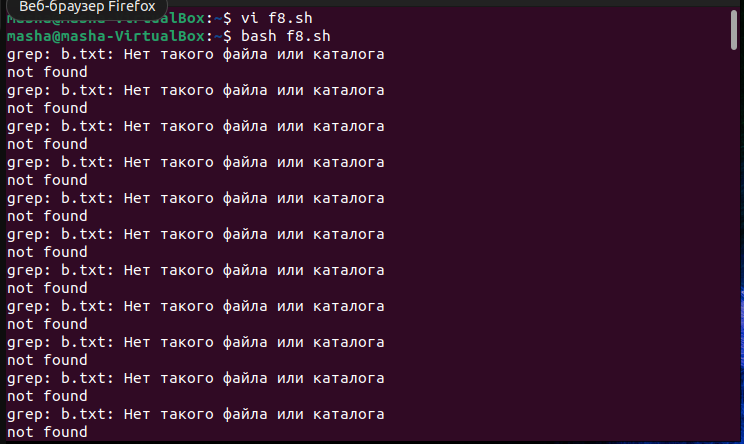
1. Циклическое чтение системного времени и очистка экрана в заданный момент.

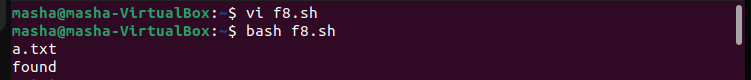


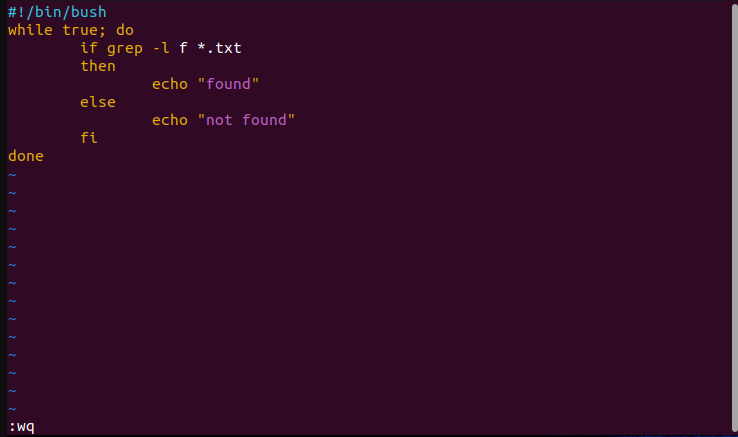




1. Циклический просмотр списка файлов и выдача сообщения при появлении заданного имени в списке.







**Ответы на контрольные вопросы**

1. *Какое назначение имеют shell - файлы?*

Интерпретатор командной строки, или shell (shell — оболочка) — эта та программа, которая принимает команды от пользователя и исполняет их.

К функциям оболочки относятся:

* Взаимодействие с пользователем (редактирование командной строки, история команд и т.д.)
* Обработка (расширение) шаблонов имен ("\*", "?" и т.д.)
* Перенаправление ввода/вывода команд
* Управление заданиями

Кроме того, shell — это специализированный язык программирования, в котором есть переменные, конструкции while, if, for и т.д., функции и много чего еще. Он позволяет писать как несложные сценарии для автоматизации повседневных задач, так и довольно сложные программы (например, запуск и останов большинства Unix'ов производятся сценариями на языке shell).

Выполняемыми [могут быть не только файлы](https://topuch.ru/ant-dlya-upravleniya-naimenovaniyami-urovnej-kategorij-peremen/index.html), полученные в результате компиляции и сборки, но и файлы, написанные на языке shell. Их выполнение происходит в режиме интерпретации с помощью shell-интерпретатора.

1. *Как создать shell - файл и сделать его выполняемым?*

Если вы подготовите в редакторе файл "f1", то это будет обычный текстовый файл, содержащий команду "echo", которая при выполнении выдает все написанное правее ее на экран. Можно сделать файл "f1" выполняемым с помощью команды "chmod 755 f1". Но его можно выполнить, вызвав явно команду (!) "sh" ("shell"): sh f1 или sh < f1

Файл можно выполнить и в текущем экземпляре "shell". Для этого существует специфическая команда "." (точка), т.е.. f1.

1. *Какие типы переменных используются в shell - файлах?*

В shell всего два типа данных: строка символов и текстовый файл

Примеры типов переменных:  
-date  
-read, echo –n  
-var echo

1. *В чем заключается анализ цепочки символов?*

Алфавит – это конечное множество I элементов, называемых символами.

Цепочка или слово в алфавите I – это конечная последовательность элементов (символов) из алфавита I. Например, если алфавит языка состоит только из заглавных и строчных букв латинского алфавита, то любые последовательности этих букв являются цепочками (словами), в том числе цепочка может состоять из одного символа.

С цепочками (словами) могут быть проделаны действия, которые имеют следующие обозначения:

* - хn. Цепочка символов х повторяется (пишется без пробелов одна за другой) n раз. Например, abba2 – это abbaabba.
* - хk. Цепочка символов х записывается в обратной последовательности. Например, portR - это trop.
* - xy. За цепочкой символов x без пробела помещается цепочка символов y.
* - х\*. Цепочка символов х в цикле может повторяться нуль и более раз.
* - х+. Цепочка символов х должна повторяться один и больше раз. В алгоритмическом языке Pascal это реализуется оператором repeat, а в языке Си – оператором цикла do while.
* - |х|. Определение длины цепочки символов х (количество символов в цепочке).
* - {} или l, или e - обозначение пустой цепочки символов.
* - [х]. Так обозначается необязательная цепочка символов. Например, такая запись нужна для того, чтобы обозначить, что перед числом знак может быть, а может и отсутствовать.

Кроме алфавита и цепочки символов (слов), важным понятием является язык.

Язык в алфавите I – это произвольное множество цепочек (слов)

1. *Какие встроенные команды используются в shell - файлах?*

Командный интерпретатор в среде UNIX выполняет две основные функции:

* представляет [интерактивный интерфейс с пользователем](https://topuch.ru/osnovi-setej-soderjanie/index.html), т.е. выдает приглашение, и обрабатывает вводимые пользователем команды;
* обрабатывает и исполняет текстовые файлы, содержащие команды интерпретатора (командные файлы);

В последнем случае, операционная система позволяет рассматривать командные файлы как разновидность исполняемых файлов. Соответственно различают два режима работы интерпретатора: интерактивный и командный.

В среде UNIX (в отличие, скажем, от DOS) имеются несколько различных командных интерпретаторов. Перечислим наиболее популярные:

* /bin/sh - Bourne shell. Исторически это первая командная оболочка, разработанная для первой версии ОС UNIX. В настоящее время эта оболочка является основной в версиях UNIX System V.
* /bin/csh - С-shell. Оболочка, синтаксис командного языка которой приближен к языку C. Является основной оболочкой для Берклеевской разновидности ОС UNIX.
* /bin/ksh - k-shell.
* /bin/rsh - Restricted shell. Представляет собой sh с ограниченными возможностями (прежде всего для защиты ОС от несанкционированных действий пользователя).

1. *Как производится управление программами?*

Когда пользователи входят в систему UNIX, они общаются с одним из интерпретаторов команд. В этой главе изучается интерпретатор команд Shell, sh. Этот интерпретатор является программой UNIX, которая поддерживает обладающий большими возможностями командный язык. Каждый вызов этого интерпретатора называется оболочкой Shell; каждая оболочка выполняет единственную функцию - считывает со стандартного устройства ввода команду и выполняет ее.

Поскольку Shell дает пользователю возможность общаться с операционной системой на языке высокого уровня, UNIX может выполнять задачи, [недоступные менее сложным](https://topuch.ru/chto-yavlyaetsya-predmetom-izucheniya-himii/index.html) операционным системам. Команды, которые обычно должны писаться на традиционных языках программирования, здесь могут быть записаны в нескольких строчках процедуры Shell. В других операционных системах команды выполняются в прямой последовательности.

В UNIX благодаря Shell команды могут:

* объединяться для образования новых команд;
* передавать позиционные параметры;
* добавляться или переименоваться пользователем;
* выполняться внутри циклов или по определенному условию;
* создаваться для локального выполнения без риска вступления в конфликт с командами других пользователей;
* выполняться в фоновом режиме.

Более того, команды могут перенаправлять ввод исходных данных для команды от одного источника к [другому и перенаправлять вывод в файл](https://topuch.ru/muletipleksirovannij-vvodvivod-bazovie-koncepcii-fajl--fundame/index.html), на терминал, принтер или другой команде. Это обеспечивает необходимую гибкость при выполнении конкретных задач.

1. *Назовите операторы создания циклов.*

Операторы создания циклов:

* For - оператор цикла с перечислением (for … do … done).
* While - оператор цикла с истинным условием (while … do … done).
* Case - оператор цикла с ложным условием (until … do … done).