



Министерство науки и высшего образования Российской Федерации
Калужский филиал
федерального государственного бюджетного
образовательного учреждения высшего образования
«Московский государственный технический университет имени Н.Э. Баумана
(национальный исследовательский университет)»
(КФ МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ ИУК «Информатика и Управление»

КАФЕДРА ИУК4 «Программное обеспечение ЭВМ, информационные технологии»

ДОМАШНЯЯ РАБОТА №1

ДИСЦИПЛИНА: «Операционные системы»

Выполнил: студент гр. ИУК4-52Б _____ (Калашников А. С.)
(Подпись) (Ф.И.О.)

Проверил: _____ (Красавин Е. В.)
(Подпись) (Ф.И.О.)

Дата сдачи (защиты):

Результаты сдачи (защиты):

- Балльная оценка:

- Оценка:

Калуга, 2022

Целью выполнения домашней работы является получение практических навыков по написанию Shell-сценариев для ОС Linux.

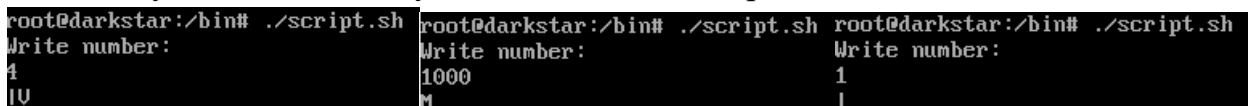
Основными задачами выполнения домашней работы являются:

1. Самостоятельно изучить синтаксис и важнейшие структуры Shell-сценариев.
2. Научиться применять, Shell-сценарии для администрирования системы.
3. Закрепить полученные в ходе выполнения лабораторных работ навыки.

Задание

Вариант 7

Написать скрипт для преобразования десятичных чисел в римскую систему счисления. Результат вывести на экран



The image shows three side-by-side terminal windows. Each window has a prompt 'root@darkstar:/bin#' and runs the command './script.sh'. The first window shows the prompt 'Write number:' followed by the input '4' and the output 'IV'. The second window shows the prompt 'Write number:' followed by the input '1000' and the output 'M'. The third window shows the prompt 'Write number:' followed by the input '1' and the output 'I'.

Рис.1 Ввод чисел

Вывод: в ходе выполнения лабораторной работы были получены практические навыки по установке и запуску ОС Linux.

КОНТРОЛЬНЫЕ ВОПРОСЫ И ЗАДАНИЯ

1. Опишите назначение Shell-скриптов.

Shell-скрипты очень хорошо подходят для быстрого создания прототипов сложных приложений, даже не смотря на ограниченный набор языковых конструкций и определенную "медлительность". Такой метод позволяет детально проработать структуру будущего приложения, обнаружить возможные "ловушки" и лишь затем приступить к написанию кода на C, C++, Java, или Perl.

2. Опишите термин BASH.

Название BASH — это аббревиатура от "Bourne-Again Shell" и игра слов от, ставшего уже классикой, "Bourne Shell" Стефена Бурна (Stephen Bourne). В последние годы BASH достиг такой популярности, что стал стандартной командной оболочкой для многих разновидностей UNIX. Большинство принципов программирования на BASH одинаково хорошо применимы и в других командных оболочках, таких как Korn Shell (ksh), от которой Bash позаимствовал некоторые особенности, а также C Shell и его производные

3. Опишите назначение символа «#», и приведите примеры его использования.

Комментарии. Строки, начинающиеся с символа # (за исключением комбинации #!) -- являются комментариями

4. Опишите для чего необходимо экранирование символов.

Экранированный символ # в операторе echo не воспринимается как начало комментария. Более того, он может использоваться в операциях подстановки параметров и в константных числовых выражениях.

5. Назовите различия одинарных и двойных кавычек.

Двойные кавычки. В строке "STRING", ограниченной двойными кавычками, не выполняется интерпретация большинства служебных символов, которые могут находиться в строке.

Одинарные кавычки. 'STRING' экранирует все служебные символы в строке STRING. Это более строгая форма экранирования.

6. Опишите понятие переменная.

Переменные — это одна из основ любого языка программирования. Они участвуют в арифметических операциях, в синтаксическом анализе строк и совершенно необходимы для абстрагирования каких-либо величин с помощью символических имен. Физически переменные представляют собой ни что иное как участки памяти, в которые записана некоторая информация

7. Приведите пример кода с условными операторами.

```
If ((num>100))then  
Else  
Fi
```

8. Опишите принцип работы с переменными в Shell.

Когда интерпретатор встречает в тексте сценария имя переменной, то он вместо него подставляет значение этой переменной. Поэтому ссылки на переменные называются подстановкой переменных. Необходимо всегда

помнить о различиях между именем переменной и ее значением. Если `variable1` — это имя переменной, то `$variable1` — это ссылка на ее значение. "Чистые" имена переменных, без префикса `$`, могут использоваться только при объявлении переменной, при присваивании переменной некоторого значения, при удалении (сбросе), при экспорте и в особых случаях - когда переменная представляет собой название сигнала. Присваивание может производиться с помощью символа `=` (например, `var1=27`), инструкцией `read` и в заголовке цикла (`for var2 in 1 2 3`).

9. Приведите пример кода с оператором цикла `while`.

`while ["$var0" -lt "$LIMIT"] do echo -n "$var0 " # -n подавляет перевод строки. var0=`expr $var0 + 1`# допускается var0=$((var0+1)). done`

10. Опишите понятие внутренней команды.

Внутренняя команда — это команда, которая встроена непосредственно в `Bash`.

11. Назовите команды для вывода информации.

`echo` выводит (на `stdout`) выражение или содержимое переменной.

`printf` - команда форматированного вывода, расширенный вариант команды `echo` и ограниченный вариант библиотечной функции `printf()` в языке C, к тому же синтаксис их несколько отличается друг от друга.

`read` "Читает" значение переменной с устройства стандартного ввода - `stdin`, в интерактивном режиме это означает клавиатуру.

12. Опишите понятие `sha-bang`.

`sha-bang` определяет интерпретатор, который вызывается для исполнения сценария, это может быть командная оболочка (`shell`), иной интерпретатор или утилита

13. Назовите способы запуска сценария

Сценарий можно вызвать, просто напечатав название файла в командной строке и нажав клавишу `Enter`