Лабораторная работа № 12

Программирование в командном процессоре ОС UNIX. Расширенное программирование

Козлов Всеволод Павлович

Содержание

# 1 Цель работы

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

# 2 Выполнение лабораторной работы

Создал и запустил командный файл, реализующий упрощённый механизм семафоров (код ниже) (рис. [[1](#fig:001)]).

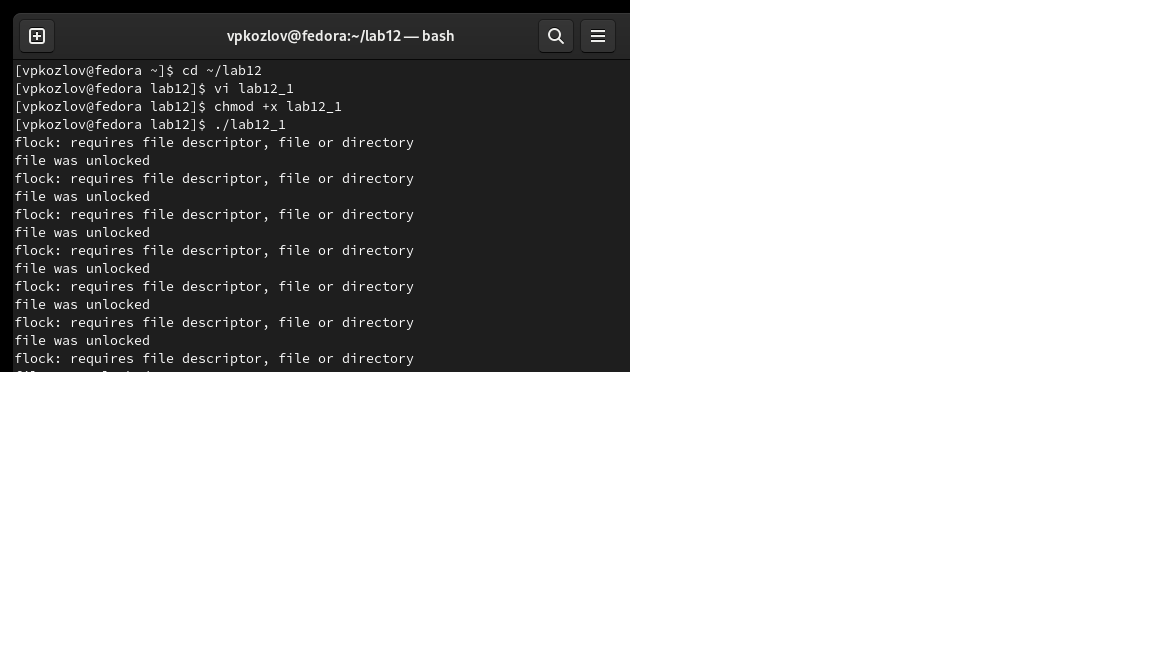


Figure 1: Создание и запуск файла\_1

Код командного файла, реализующего упрощённый механизм семафоров (рис. [[2](#fig:002)]).

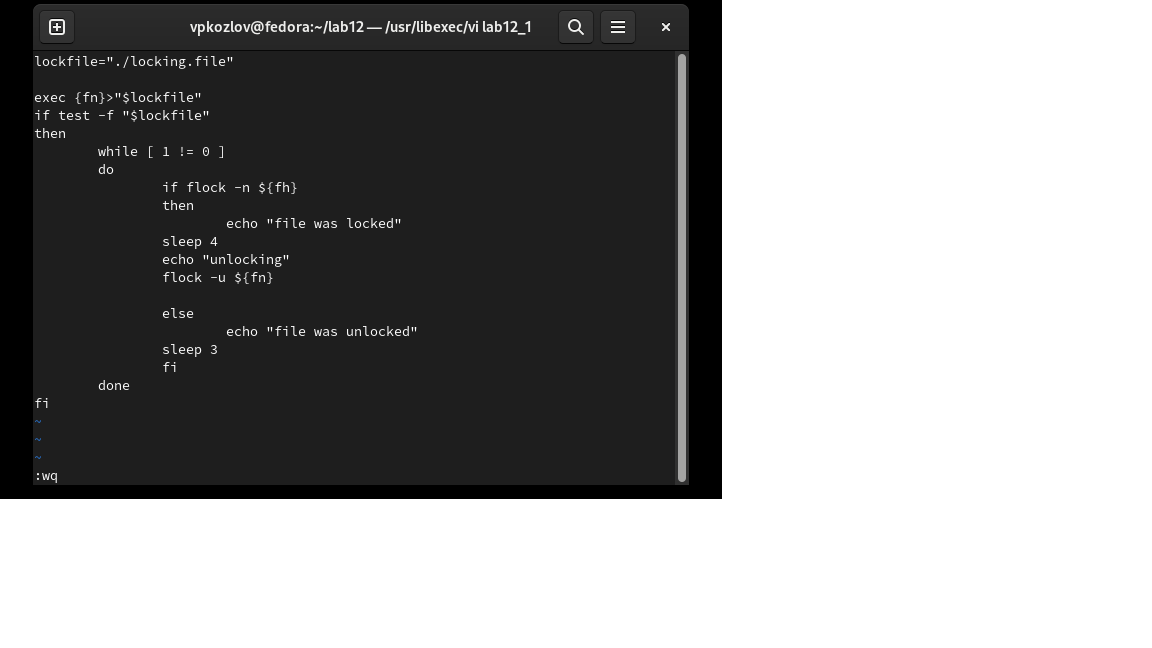


Figure 2: Код файла\_1

Создал и запустил командный файл, реализующий команду man (код ниже) (рис. [[3](#fig:003)]).

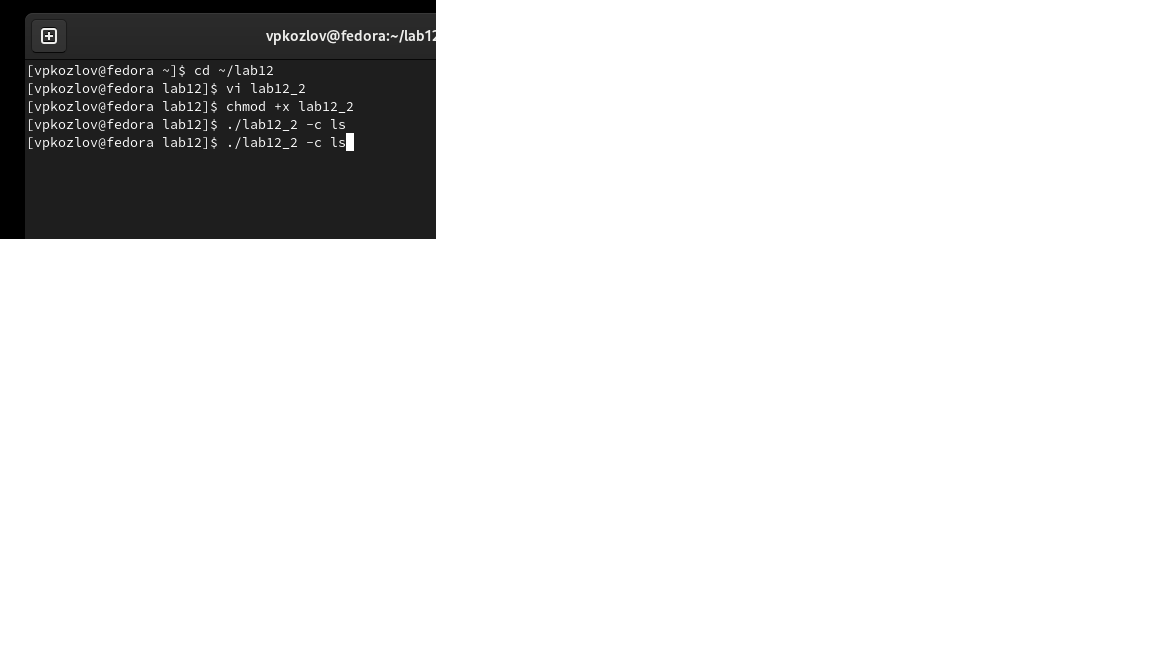


Figure 3: Создание и запуск файла\_2

Код командного файла, реализующего команду man (рис. [[4](#fig:004)]).

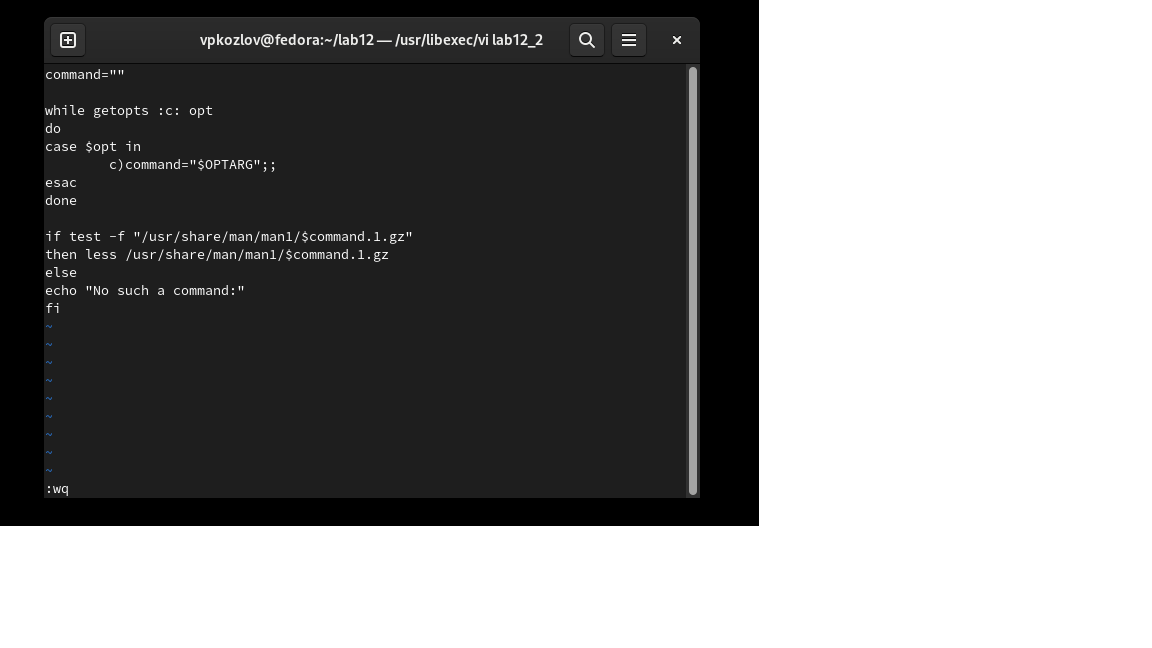


Figure 4: Код файла\_2

Демонстрация работы командого файла\_2 (рис. [[5](#fig:005)]).

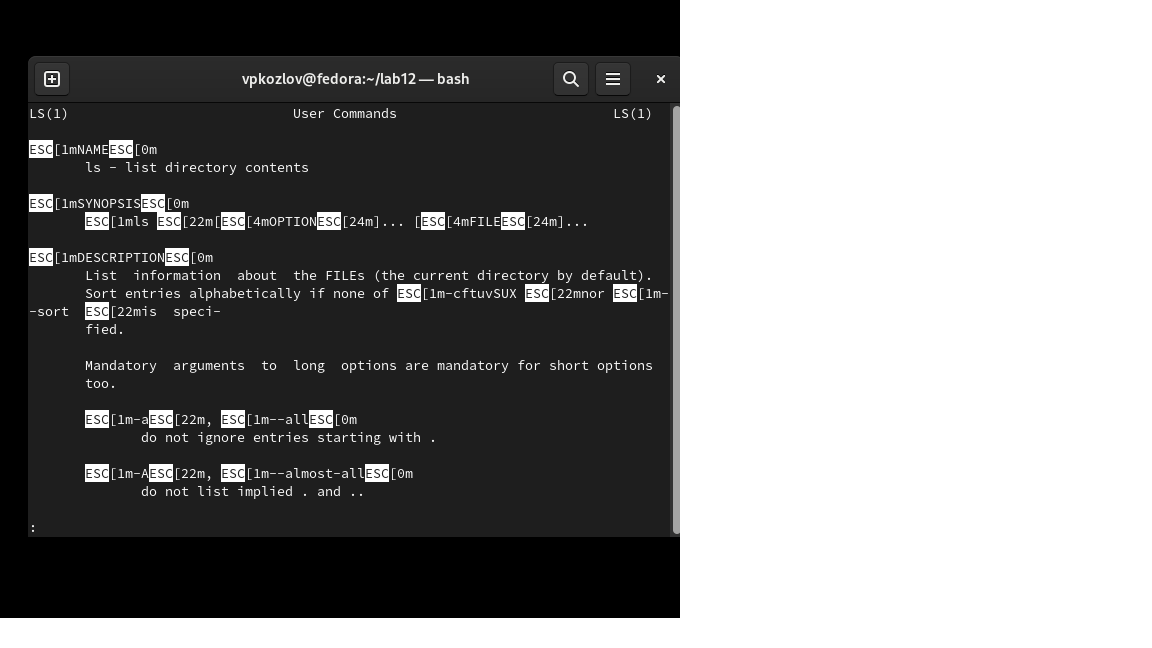


Figure 5: Работа файла\_2

Создал и запустил командный файл, генерирующий случайную последовательность букв латинского алфавита (код ниже) (рис. [[6](#fig:006)]).

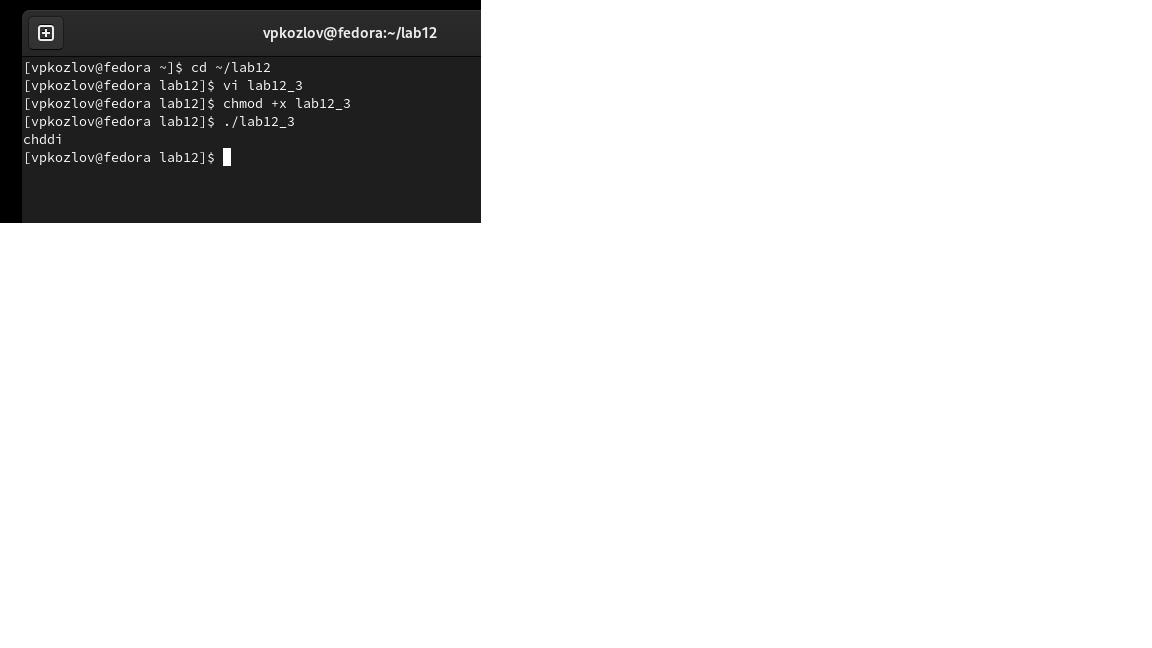


Figure 6: Создание и запуск файла\_3

Код командного файла, генерирующего случайную последовательность букв латинского алфавита (рис. [[7](#fig:007)]).

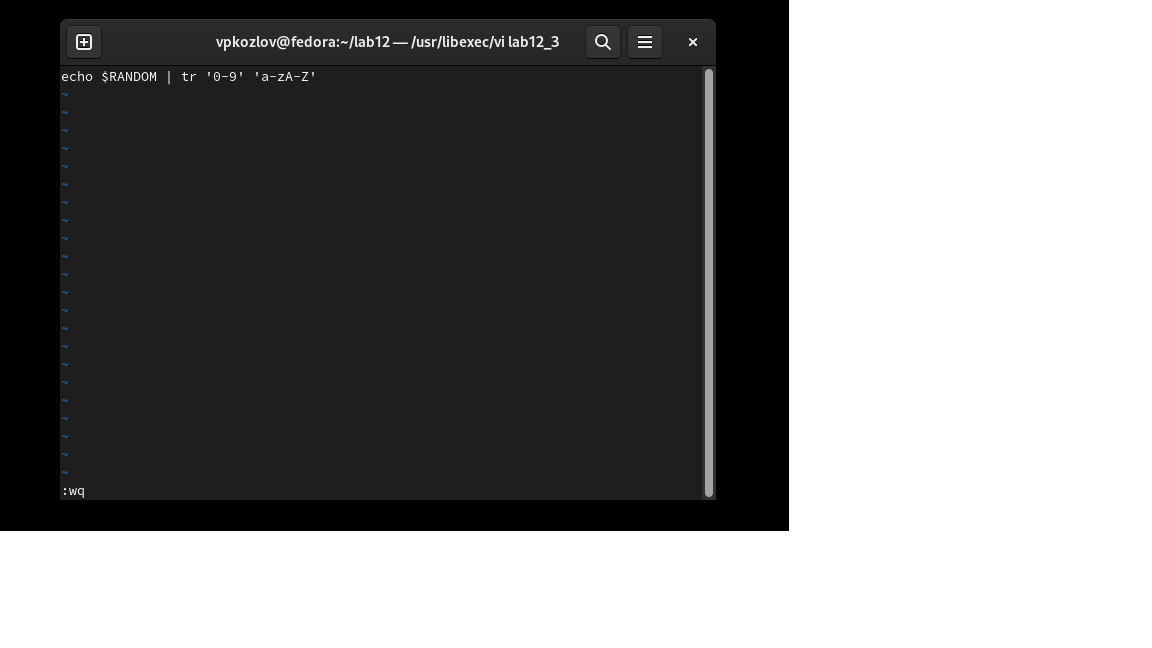


Figure 7: Код файла\_3

# 3 Выводы

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

# 4 Ответы на контрольные вопросы

1. Найдите синтаксическую ошибку в следующей строке: 1 while [$1 != “exit”] В данной строчке допущены следующие ошибки: не хватает пробелов после первой скобки [ и перед второй скобкой ] выражение $1 необходимо взять в “”, потому что эта переменная может содержать пробелы Таким образом, правильный вариант должен выглядеть так: while [ “$1” != “exit” ]
2. Как объединить (конкатенация) несколько строк в одну? Чтобы объединить несколько строк в одну, можно воспользоваться несколькими способами: Первый: VAR1=“Hello,” VAR2=” World” VAR3=“VAR1VAR2” echo “$VAR3" Результат: Hello, World Второй: VAR1="Hello, " VAR1+=" World" echo "$VAR1” Результат: Hello, World
3. Найдите информацию об утилите seq. Какими иными способами можно реализовать её функционал при программировании на bash? Команда seq в Linux используется для генерации чисел от ПЕРВОГО до ПОСЛЕДНЕГО шага INCREMENT. Параметры: seq LAST: если задан только один аргумент, он создает числа от 1 до LAST с шагом шага, равным 1. Если LAST меньше 1, значение is не выдает. seq FIRST LAST: когда заданы два аргумента, он генерирует числа от FIRST до LAST с шагом 1, равным 1. Если LAST меньше FIRST, он не выдает никаких выходных данных. seq FIRST INCREMENT LAST: когда заданы три аргумента, он генерирует числа от FIRST до LAST на шаге INCREMENT. Если LAST меньше, чем FIRST, он не производит вывод. seq -f «FORMAT» FIRST INCREMENT LAST: эта команда используется для генерации последовательности в форматированном виде. FIRST и INCREMENT являются необязательными. seq -s «STRING» ПЕРВЫЙ ВКЛЮЧЕНО: Эта команда используется для STRING для разделения чисел. По умолчанию это значение равно /n. FIRST и INCREMENT являются необязательными. seq -w FIRST INCREMENT LAST: эта команда используется для выравнивания ширины путем заполнения начальными нулями. FIRST и INCREMENT являются необязательными.
4. Какой результат даст вычисление выражения $((10/3))? Результатом данного выражения $((10/3)) будет 3, потому что это целочисленное деление без остатка.
5. Укажите кратко основные отличия командной оболочки zsh от bash. Отличия командной оболочки zsh от bash: В zsh более быстрое автодополнение для cd с помощью Тab В zsh существует калькулятор zcalc, способный выполнять вычисления внутри терминала В zsh поддерживаются числа с плавающей запятой В zsh поддерживаются структуры данных «хэш» В zsh поддерживается раскрытие полного пути на основенеполных данных В zsh поддерживается замена части пути В zsh есть возможность отображать разделенный экран, такой же как разделенный экран vim
6. Проверьте, верен ли синтаксис данной конструкции 1 for ((a=1; a <= LIMIT; a++)) for ((a=1; a <= LIMIT; a++)) синтаксис данной конструкции верен, потому что, используя двойные круглые скобки, можно не писать $ перед переменными ().
7. Сравните язык bash с какими-либо языками программирования. Какие преимущества у bash по сравнению с ними? Какие недостатки? Преимущества скриптового языка bash: - Один из самых распространенных и ставится по умолчанию в большинстве дистрибутивах Linux, MacOS - Удобное перенаправление ввода/вывода - Большое количество команд для работы с файловыми системами Linux - Можно писать собственные скрипты, упрощающие работу в Linux Недостатки скриптового языка bash: - Дополнительные библиотеки других языков позволяют выполнить больше действий - Bash не является языков общего назначения - Утилиты, при выполнении скрипта, запускают свои процессы, которые, в свою очередь, отражаются на быстроте выполнения этого скрипта - Скрипты, написанные на bash, нельзя запустить на других операционных системах без дополнительных действий

# Список литературы