Отчет по лабораторной работе 13

Дорофеева Алёна Тимофеевна

# Цель работы

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

# Задание

Проследовав плану выполнения работы, научиться писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.

# Выполнение лабораторной работы

Описываются проведённые действия, в качестве иллюстрации даётся ссылка на иллюстрацию

1. Создаем файл для кода и файл, с котором будем работать, заходим в редактор emacs. После работы в emacs необходимо изменить права доступа (рис.1)

*Рис. 1: Работа с первым файлом в строке*

1.1. Написала командный файл, реализующий упрощённый механизм семафоров. Командный файл в течение некоторого времени t1 дожидается освобождения ресурса, выдавая об этом сообщение, а дождавшись его освобождения, использует его в течение некоторого времени t2<>t1, также выдавая информацию о том, что ресурс используется соответствующим командным файлом (процессом) (рис.1.1)

*Рис. 1.1: Работа в emacs*

1.2. Запускаем скрипт. Если открыть второй терминал и запустить этот же скрипт, можно заметить, что второй терминал уже не может открыть файл, потому что этот файл работает в другом процессе (рис.1.2.1). Если закрыть первый терминал, то второй получит доступ к файлу (рис.1.2.2)

*Рис. 1.2.1: Проверяем работу написанной программы*

*Рис. 1.2.2: Продолжение работы скрипта*

1. Создаем второй файл для работы с ним, открываем emacs и меняем права доступа (рис.2)

*Рис. 2: Работа со вторым файлом в строке*

2.1. Пишем программу, реализующую команду man. Нам необходимо изучить содержимое каталога /usr/share/man/man1. В нем находятся архивы текстовых файлов, содержащих справку по большинству установленных в системе программ и команд. Каждый архив можно открыть командой less сразу же просмотрев содержимое справки (рис.2.1)

*Рис. 2.1: Написание программы в emacs*

2.2 Запускаем файл, проверяем работу программы, команда cd не имеет справки (рис.2.2)

*Рис. 2.2: Проверка работы написанной программы*

2.3 Запускаем снова, проверяем на команде ls. (рис.2.3.1) Получаем справку по этой команде (рис. 2.3.2)

*Рис. 2.3.1: Запускаем файл*

*Рис. 2.3.2: Получаем справку по команде ls*

1. Создаем третий файл, запускаем emacs и меняем права доступа (рис. 3)

*Рис. 3: Работа в строке с третьим файлом*

3.1. Используя встроенную переменную $RANDOM, написала командный файл, генерирующий случайную последовательность букв латинского алфавита (рис.3.1)

*Рис. 3.1: Работа с программой в emacs*

3.2. Проверяем работу команды в строке (рис. 3.2)

*Рис. 3.2.1: Запуск программы в строке*

# Вывод

Я изучила основы программирования в оболочке ОС UNIX. Научилась писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.

Библиографический список:

[Инструкции к лабораторной работе (файл 008-lab\_shell\_prog\_1.pdf).](https://esystem.rudn.ru/pluginfile.php/1142377/mod_resource/content/2/008-lab_shell_prog_1.pdf)

[Инструкции к лабораторной работе (файл 009-lab\_shell\_prog\_2.pdf)](https://esystem.rudn.ru/pluginfile.php/1142380/mod_resource/content/3/009-lab_shell_prog_2.pdf)

[Инструкции к лабораторной работе (файл 010-lab\_shell\_prog\_3.pdf)](https://esystem.rudn.ru/pluginfile.php/1142383/mod_resource/content/2/010-lab_shell_prog_3.pdf)