Отчет по лабораторной работе №11

Программирование в командном процессоре ОС UNIX. Ветвления и циклы

Кочкарев “sakochkarev” Станислав

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 Задание

1. Написать командный файл, реализующий упрощенный механизм семафоров.
2. Написать командный файл, аналог команды man.
3. Написать командный файл, генерирующий случайную строку.

# 3 Выполнение лабораторной работы

Первым заданием было написание командного файла, реализующего упрощенный механизм семафоров.

Ниже приведен листинг итогового командного файла (рис. 6), а также результат его работы (рис. 7).



Рис. 1: Листинг командного файла

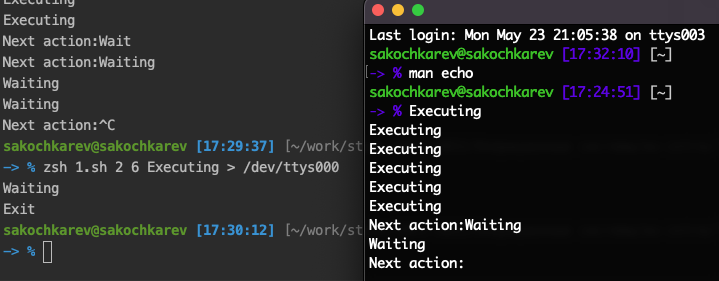


Рис. 2: Выполнение команды

Следующим заданием было написание командного файла, который альтернирует поведение команды man. Командный файл принимает один аргумент – название команды, мануал которой пользователь хочет просмотреть. Для работы командного файла была использована директория /usr/share/man/man1, в которой происходил поиска мануала приведенной команды. Далее мануал выводился с помощью команды less.

Ниже приведен листинг итогового командного файла (рис. 3), а также результат его работы (рис. 4, 5).



Рис. 3: Листинг командного файла

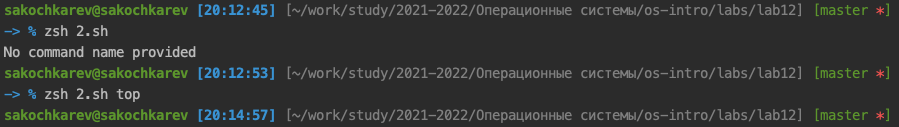


Рис. 4: Выполнение команды

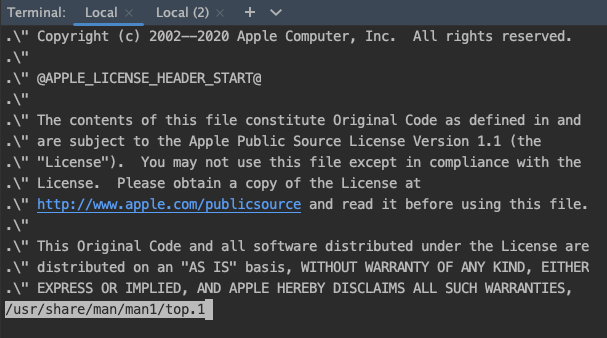


Рис. 5: Выполнение команды

Последним заданием было написание командного файла, который, используя встроенную переменную $RANDOM выводит случайную последовательности латинских букв.

Ниже приведен листинг итогового командного файла (рис. 6), а также результат его работы (рис. 7).



Рис. 6: Листинг командного файла

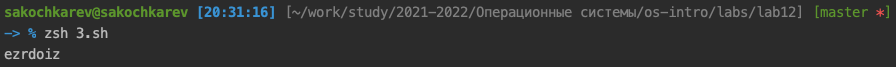


Рис. 7: Выполнение команды

# 4 Выводы

По выполнении лабораторной работы мы изучили основы программирования в оболочке ОС UNIX, а также научились писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.

# 5 Контрольные вопросы

1. Синтаксическая ошибка заключается в отсутствии пробелов вокруг выражения в []. Правильная строка:

while [ $1 != "exit" ]

1. Можно использовать подстановку переменных в строку. Например:

string1="hello"  
string2="world"  
result="${string1} ${world}"

1. Подобный seq функционал можно реализовать используя формат (()) вместе с Си-подобным итерированием, например:

for ((i=0; i<10; i++)); do  
 echo $i;  
done

1. 3.
   * Прокаченное авто-дополнение
   * Переход без cd
   * Поправление ошибок
   * Переходы по истории
2. Да, синтаксис верен.
3. Сравнивая bash, например, с высокоуровневым языком программирования Python, bash уступает во многих сферах и применениях, так как это не нацеленный на разработку ПО язык, однако в среде UNIX, особенно в коммандных файлах (скриптах), ему практически нет равных.