

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ
Факультет физико-математических и естественных наук
Кафедра прикладной информатики и теории вероятностей

ОТЧЕТ

ПО ЛАБОРАТОРНОЙ РАБОТЕ №6

дисциплина:Операционные системы

Студент: Мартемьянов Александр

Группа: НПМбв-02-18

МОСКВА

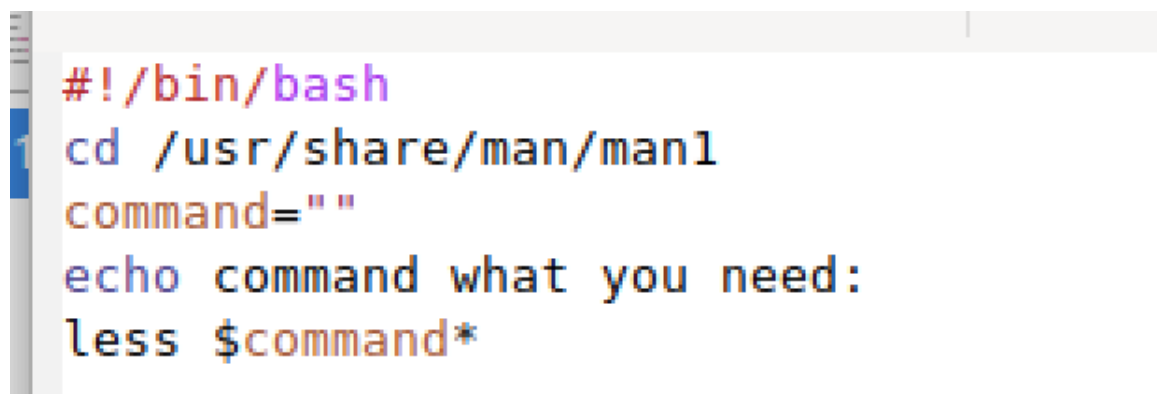
2022 г.

2)Задание

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

3) Последовательность выполнения работы

1. Написать командный файл, реализующий упрощённый механизм семафоров. Командный файл должен в течение некоторого времени t_1 дожидаться освобождения ресурса, выдавая об этом сообщение, а дождавшись его освобождения, использовать его в течение некоторого времени $t_2 < t_1$, также выдавая информацию о том, что ресурс используется соответствующим командным файлом (процессом). Запустить командный файл в одном виртуальном терминале в фоновом режиме, перенаправив его вывод в другой $> /dev/tty\#$, где $\#$ — номер терминала куда перенаправляется вывод, в котором также запущен этот файл, но не в фоновом, а в привилегированном режиме. Доработать программу так, чтобы имелась возможность взаимодействия трёх и более процессов
2. Реализовать команду `man` с помощью командного файла. Изучите 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. Командный файл должен получать в виде аргумента командной строки название команды и в виде результата выдавать справку об этой команде или сообщение об отсутствии справки, если соответствующего файла нет в каталоге `man1`



```
#!/bin/bash
cd /usr/share/man/man1
command=""
echo command what you need:
less $command*
```

Рис 3. 3 «Реализация команды `man` с помощью командного файла»

```
Файл  Правка  Вид  Поиск  Терминал  Справка
.\\" DO NOT MODIFY THIS FILE!  It was generated by help2man 1.47.3.
.TH TEST "1" "September 2019" "GNU coreutils 8.30" "User Commands"
.SH NAME
test \- check file types and compare values
.SH SYNOPSIS
.B test
.I EXPRESSION
.br
.B test
.br
.\" \& tells docclifter the brackets are literal (Bug#31803).
.B [\&
.I EXPRESSION
.B ]\&
.br
.B "[\& ]\&"
.br
.B [\&
.I OPTION
.SH DESCRIPTION
.\" Add any additional description here
.PP
Exit with the status determined by EXPRESSION.
.TP
\fb\-\-help\fr
display this help and exit
.TP
\fb\-\-version\fr
output version information and exit
.PP
An omitted EXPRESSION defaults to false.  Otherwise,
EXPRESSION is true or false and sets exit status.  It is one of:
.TP
( EXPRESSION )
EXPRESSION is true
.TP
! EXPRESSION
EXPRESSION is false
.TP
EXPRESSION1 \fb\-a\fr EXPRESSION2
both EXPRESSION1 and EXPRESSION2 are true
[.1.gz (file 1 of 1846)]
```

Рис 3. 4 «Реализация команды man с помощью командного файла»

- Используя встроенную переменную \$RANDOM, напишите командный файл, генерирующий случайную последовательность букв латинского алфавита. Учтите, что \$RANDOM выдаёт псевдослучайные числа в диапазоне от 0 до 32767.

```
#!/bin/bash
word=
echo number-list of words:
read word
for((i=0;i<$word;i++))
do echo $RANDOM | tr '[a-z]'
done
```

Рис 3. 5 «Генерация случайного числа»

```
user@user-VirtualBox:~/work/os/lab13$ ./lab13-3.sh
number-list of words:
5
184e4
95ee
dlaa
abel8
baecd
user@user-VirtualBox:~/work/os/lab13$ ./lab13-3.sh
number-list of words:
10
b9bc4
1b8be
dca4
5a9e
b99c9
bee58
5a98
1c595
b18c
b99c1
user@user-VirtualBox:~/work/os/lab13$
```

Рис 3. 6 «Генерация случайного числа»

4) Выводы согласованные с заданием работы

В результате выполнения этой работы изучили основы программирования в оболочке ОС UNIX. Научились писать более сложные командные файлы с использованием логических управляющих конструкций и циклов

5) Ответы на контрольные вопросы

1. Найдите синтаксическую ошибку в следующей строке: while [\$1 != "exit"]

Для правильного исполнения командной строки while [\$1 != "exit"] квадратные скобки нужно поменять на круглые скобки

2. Как объединить (конкатенация) несколько строк в одну?

Использовать знак \$ между двумя переменными символьного типа для объединения нескольких строк в одно единую целую строку

3. Найдите информацию об утилите seq. Какими иными способами можно реализовать её функционал при программировании на bash?

seq - выводит последовательность целых или действительных чисел для передачи в другие программы, на языке bash её можно реализовать с помощью цикла for

4. Какой результат даст вычисление выражения \$((10/3))?

$10 / 3 = 3$ (дробная часть будет отброшена, так как числа целые, но если бы было $10.000/3.000 = 3.333$)

5. Укажите кратко основные отличия командной оболочки zsh от bash.

Zsh выполнен на основе bash, а поэтому наследует как недостатки так и преимущества bash, но в zsh есть автодополнение, горячие клавиши, алиасы, различные удобства в виде тем, плагинов и расширенной поддержке, однако есть минус всей этой красоты и удобства, на узкоспециализированных машинах это красоты быть не может в принципе из-за ограничений железа или системного администратора

6. Проверьте, верен ли синтаксис данной конструкции for ((a=1; a <= LIMIT; a++))

Синтаксис верен, однако для надежности лучше писать for и ((some words)) без пробела, т.е. слитно

7. Сравните язык bash с какими-либо языками программирования. Какие преимущества у bash по сравнению с ними? Какие недостатки?

Плюсы Python - является мощным ООП-языком программирования, проблем при написании кода будет меньше, имеет автозаполнение, улучшенного синтаксиса, кроссплатформенности, подходит для автоматизирования процессов разработки ПО

Минусы Python - скорость, недостаток в скорости ограничивает область применения этого языка в задачах

Плюсы Bash - является низкоуровневым языком программирования, что позволяет писать скрипты приближенные к синтаксису C-language, подходит больше для системного администрирования

Минусы Bash - не является кроссплатформенным языком для написания скриптов