

Лабораторная работа №12

Макарова Анастасия Михайловна

Цель работы

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

Первая программа

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)

```
#!/bin/bash
t1=$1
t2=$2
s1=$(date +%s)
s2=$(date +%s)
((t=$s2-$s1))
while ((t<t1))
do
    echo "Ожидание"
    sleep 1
    s2=$(date +%s)
    ((t=$s2-$s1))
done
s1=$(date +%s)
s2=$(date +%s)
((t=$s2-$s1))
while ((t<t2))
do
    echo "Выполнение"
    sleep 1
    s2=$(date +%s)
    ((t=$s2-$s1))
done
█

U:***- semafor.sh Top L24 (Shell-script[sh])
```

Написание программы

```
[ammakarova@10 ~]$ emacs &  
[1] 5474  
[ammakarova@10 ~]$ chmod +x semafor.sh  
[ammakarova@10 ~]$ ./semafor.sh 3 7  
Ожидание  
Ожидание  
Ожидание  
Выполнение  
Выполнение  
Выполнение  
Выполнение  
Выполнение  
Выполнение  
Выполнение  
[ammakarova@10 ~]$
```

Работа программы

Вторая программа

Реализовать команду `man` с помощью командного файла. 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
a=$1
if [ -f /usr/share/man/man1/$a.1.gz ]
then
    gunzip -c /usr/share/man/man1/$1.1.gz | less
else
    echo "Нет справки по данной команде"
fi
```

U:--- **man.sh** All L1 (Shell-script[bash])

Написание программы

```
[ammakarova@10 ~]$ chmod +x man.sh  
[ammakarova@10 ~]$ ./man.sh mkdir  
[ammakarova@10 ~]$ ./man.sh cat  
[ammakarova@10 ~]$ ./man.sh rm  
[ammakarova@10 ~]$ ./man.sh ls
```

Команды в консоли


```

.\" DO NOT MODIFY THIS FILE! It was generated by help2man 1.47.3.
.TH MKDIR "1" "March 2022" "GNU coreutils 8.32" "User Commands"
.SH NAME
mkdir \- make directories
.SH SYNOPSIS
.B mkdir
[\fI\,OPTION\/\fR]... \fI\,DIRECTORY\/\fR...
.SH DESCRIPTION
.\" Add any additional description here
.PP
Create the DIRECTORY(ies), if they do not already exist.
.PP
Mandatory arguments to long options are mandatory for short options too.
.TP
\fB\m\fR, \fB\-\mode\fR=\fI\,MODE\/\fR
set file mode (as in chmod), not a=rwx \- umask
.TP
\fB\p\fR, \fB\-\parents\fR
no error if existing, make parent directories as needed
.TP
\fB\v\fR, \fB\-\verbose\fR
print a message for each created directory
.TP
\fB\Z\fR
set SELinux security context of each created directory
to the default type
.TP
\fB\-\context\fR[=\fI\,CTX\/\fR]
like \fB\Z\fR, or if CTX is specified then set the SELinux
or SMACK security context to CTX
.TP
:

```

Работа программы

Третья программа

Используя встроенную переменную `$RANDOM`, написать командный файл, генерирующий случайную последовательность букв латинского алфавита.

```
#!/bin/bash
a=$1
for ((i=0; i<$a; i++))
do
    ((char=$RANDOM%26+1))
    case $char in
        1) echo -n a;;
        2) echo -n b;;
        3) echo -n c;;
        4) echo -n d;;
        5) echo -n e;;
        6) echo -n f;;
        7) echo -n g;;
        8) echo -n h;;
        9) echo -n i;;
        10) echo -n j;;
        11) echo -n k;;
        12) echo -n l;;
        13) echo -n m;;
        14) echo -n n;;
        15) echo -n o;;
        16) echo -n p;;
        17) echo -n q;;
        18) echo -n r;;
        19) echo -n s;;
        20) echo -n t;;
        21) echo -n u;;
        22) echo -n v;;
        23) echo -n w;;
    esac
done
```

U:--- random.sh Top L28 (Shell-script[sh])

Написание программы

```
[ammakarova@10 ~]$ chmod +x random.sh
[ammakarova@10 ~]$ ./random.sh 10
cgbcsmxyhu
[ammakarova@10 ~]$ ./random.sh 39
heclnsjjldqusqiverzcrcdfifgbvffaepztkmxw
[ammakarova@10 ~]$ ./random.sh 98
jvtxokbbnhpfcoomsiutzuznwtblaglcftbkdsngxkgihaouzibuscngxhxdyywdmlywfacieqvgchfmdvkismqddrzdykeua
```

Работа программы

Выводы

- 1) Изучила основы программирования в оболочке ОС UNIX/Linux;
- 2) Научилась писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.

СПАСИБО ЗА ВНИМАНИЕ :)