

Презентация по лабораторной работе №12

Операционные системы

Сячинова Ксения Ивановна

24 марта 2023

Российский университет дружбы народов, Москва, Россия

Цель работы

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

Задание

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 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`.
3. 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.

Выполнение лабораторной работы

1. Создадим рабочее пространство для скриптов данной лабораторной работы. также создадим файл, в который запишем наш скрипт

```
kisyachinova1@dk8n78 ~/work/os $ mkdir lab12
kisyachinova1@dk8n78 ~/work/os $ cd lab12
kisyachinova1@dk8n78 ~/work/os/lab12 $ touch os12.1.sh
kisyachinova1@dk8n78 ~/work/os/lab12 $ emacs $
```

Рис. 1: Создание папки

Напишем скрипт для выполнения задания.

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

Рис. 2: Скрипт

Даём право на выполнение и проверяем работу. Скрип работает корректно.

```
kisyachinova1@dk8n78 ~/work/os/lab12 $ chmod +x os12.1.sh
kisyachinova1@dk8n78 ~/work/os/lab12 $ ./os12.1.sh 4 5
Ожидайте
Ожидайте
Ожидайте
Ожидайте
Выполнение
Выполнение
Выполнение
Выполнение
Выполнение
```

Рис. 3: Проверка

Далее изменим скрипт там, чтобы можно было его выполнять в нескольких терминалах. При проверке мне было отказано в доступе

```
done
}
t1=$1
t2=$2
command=$3
while true
do
    if [ "$command" = "Выход" ]
    then
        echo "Выход"
        exit 0
    fi
    if [ "$command" = "Ожидание" ]
    then ogidanie
    fi
    if [ "$command" = "Выполнение" ]
    then vipolnenie
    fi
    echo "Следующее действие"
    read command
done
}
```

Рис. 4: Скрипт

2. Выполним второе задание. Для начала изучим содержимое файла “/usr/share/man/man1”

```
kisyaehinoval@dk8n72 ~/work/os/lab12 $ cd
kisyaehinoval@dk8n72 ~ $ cd /usr/share/man/man1
kisyaehinoval@dk8n72 /usr/share/man/man1 $ ls
alltopps.1.bz2
7x.1.bz2
7za.1.bz2
7zr.1.bz2
a2ps.1.bz2
a52dec.1.bz2
ab.1.bz2
acllocal-1.14.1.bz2
acllocal-1.15.1.bz2
acllocal-1.16.1.bz2
aconnect.1.bz2
acyclic.1.bz2
adddebug.1.bz2
addedgeg.1.bz2
addftinfo.1.bz2
addptg.1.bz2
advdef.1.bz2
advimg.1.bz2
advpng.1.bz2
advzip.1.bz2
afm2pl.1.bz2
afm2tfo.1.bz2
afmtodit.1.bz2
afe.1.bz2
```

Рис. 6: Содержимое файла

Создадим файла для скрипта и напишем его. Далее добавим право на выполнение и проверим его.

```
#!/bin/bash
c=$1
if [ -f /usr/share/man/man1/${c}.gz ]
then
gunzip -c /usr/share/man/man1/${c}.gz | less
else
echo "Справки по данной команде нет"
fi
..
```

Рис. 7: Скрипт

```
kizyachinov@b6b6n78 ~/work/os/1ab12 $ chmod +x os12.2.sh
kizyachinov@b6b6n78 ~/work/os/1ab12 $ ./os12.2.sh ls
Справки по данной команде нет
kizyachinov@b6b6n78 ~/work/os/1ab12 $ ./os12.2.sh mkdir
Справки по данной команде нет
```

Рис. 8: Проверка

3. Напишем скрипт для выполнение третьего задания.

```
#!/bin/bash
k=$1
for (( i=0; i<$k; 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;;
        24) echo -n x;;
        25) echo -n y;;
        26) echo -n z;;
    esac
done
echo
```

```
kisyachinova1@dk8n78 ~/work/os/lab12 $ ./os12.3.sh 5  
owqea  
kisyachinova1@dk8n78 ~/work/os/lab12 $ ./os12.3.sh 10  
afwazyktea
```

Рис. 10: Проверка

Выводы

В процессе выполнения данной лабораторной работы я изучила основы программирования в оболочке ОС UNIX и научилась писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.