

Лабораторная работа №12

Программирование в командном процессоре ОС UNIX. Расширенное программирование

Захаренко А.В.

19.03.2023

Российский университет дружбы народов, Москва, Россия

Информация

- Захаренко Анастасия Викторовна
- студентка первого курса бакалавриата
- направление бизнес-информатика, факультет физико-математических и естественных наук
- Российский университет дружбы народов

Элементы презентации

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

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.

1. Я написала командный файл, реализующий упрощённый механизм семафоров

```
[avzakharenko@fedora ~]$ cd ~/work/study/2022-2023/"Операционные системы"/os-intro/labs/lab12
[avzakharenko@fedora lab12]$ touch file.sh
[avzakharenko@fedora lab12]$ chmod +x file.sh
[avzakharenko@fedora lab12]$ ./file.sh
lock
work
work
work
work
work
work
[avzakharenko@fedora lab12]$
```

Рис. 1: реализация1

```
#!/bin/bash
lockfile="./lockfile"
exec {fn}>$lockfile
echo "lock"
until flock -n ${fn}
do
    echo "not lock"
    sleep 1
    flock -n ${fn}
done
for ((i=0; i<=5; i++))
do
    echo "work"
    sleep 1
done
```

2. я реализовала команду man с помощью командного файла.

```
#!/bin/bash  
cd /usr/share/man/man1  
less $1*
```

Рис. 3: код2

```
[avzakharenko@fedora lab12]$ touch file-1.sh  
[avzakharenko@fedora lab12]$ chmod +x file-1.sh  
[avzakharenko@fedora lab12]$ ./file-1.sh  
[avzakharenko@fedora lab12]$ ./file-1.sh less
```

Рис. 4: реализация2

```
ESC[1mNAMEESC[0m
```

```
less - opposite of more
```

```
ESC[1mSYNOPSISESC[0m
```

```
ESC[1mless -?ESC[0m
```

```
ESC[1mless --helpESC[0m
```

```
ESC[1mless -VESC[0m
```

```
ESC[1mless --versionESC[0m
```

```
ESC[1mless [-[+]aABcCdeEfFgGiIJKLmMnNqQrRsSuUVwWX~]ESC[0m
```

```
ESC[1m[-b ESC[4mESC[22mspaceESC[24mESC[1m] [-h ESC[4mESC[22mlinesESC[24mESC[1m] [-j ESC[4mESC[22mlineESC[24mESC[1m] [-k ESC[4mESC[22mkeyfileESC[24mESC[1m]ESC[0m
```

```
ESC[1m[-{oO} ESC[4mESC[22mlogfileESC[24mESC[1m] [-p ESC[4mESC[22mpatESC[24mESC[1m] [-P ESC[4mESC[22mpromptESC[24mESC[1m] [-t ESC[4mESC[22mtagESC[24mESC[1m]ESC[0m
```

Рис. 5: реализация3

3. Используя встроенную переменную \$RANDOM, я написала командный файл, генерирующий случайную последовательность букв латинского алфавита.

```
#!/bin/bash
M=10
c=1
d=1
echo
echo "10 random words:"
while (($c!=($M+1)))
do
    echo $(for((i=1;i<=10;i++)); do printf '%s' "${RANDOM:0:1}"; done) | tr '0-9' '[a-z]'
    echo $d
    ((c+=1))
    ((d+=1))
done
```

Рис. 6: код3

```
[avzakharenko@fedora lab12]$ ./file-2.sh

10 random words:
dcbfcbccbb
1
ecbbdccbbi
2
bcbbccbbbb
3
bcbdcfcebb
4
bhcbddbfb
5
cbccfbccbh
6
cbcdbbcibc
7
bcdjccddb
8
icgbbbddhb
9
bgdcdcccd
10
[avzakharenko@fedora lab12]$
```

Рис. 7: реализация4

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