

Отчет по лабораторной работе №12

Операционные системы

Лукина Ирина

Содержание

1	Цель работы	5
2	Ход работы	6
3	Выводы	10
	Список литературы	11

Список иллюстраций

2.1	Код	6
2.2	Работа файла	7
2.3	Каталог	7
2.4	Файл	8
2.5	Работа файла	8
2.6	Код	8
2.7	Работа файла	9

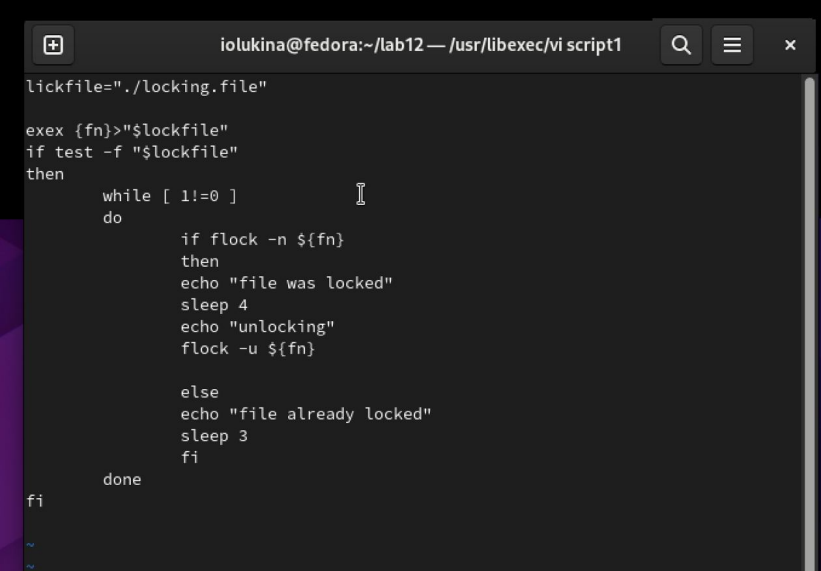
Список таблиц

1 Цель работы

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

2 Ход работы

1. Создаю файл и прописываю скрипт, реализующий упрощенный механизм семафоров и проверяю его работу в терминале.

A screenshot of a terminal window with a dark background. The window title bar shows the user 'iolukina' on a 'fedora' machine in the directory '~/lab12', editing a file named '/usr/libexec/vi script1'. The script content is as follows:

```
lockfile="./locking.file"

exex {fn}>"$lockfile"
if test -f "$lockfile"
then
    while [ 1!=0 ]
    do
        if flock -n ${fn}
        then
            echo "file was locked"
            sleep 4
            echo "unlocking"
            flock -u ${fn}

        else
            echo "file already locked"
            sleep 3
        fi
    done
fi
```

Рис. 2.1: Код

```
file was locked
unlocking
file was locked
unlocking
file was locked
unlocking
file was locked
unlocking
file was locked
unlocking
file was locked
aunlocking
file was locked

aunlocking
file was locked
unlocking
file was locked
```

Рис. 2.2: Работа файла

2. Изучаю содержимое каталога `/usr/share/man/man1`. И пишу командный файл.

```
[iolukina@fedora ~]$ cd /usr/share
[iolukina@fedora share]$ cd man
[iolukina@fedora man]$ cd man1
[iolukina@fedora man1]$ ls
.:.1.gz
' [.1.gz'
a2ping.1.gz
ab.1.gz
abrt.1.gz
abrt-action-analyze-backtrace.1.gz
abrt-action-analyze-c.1.gz
abrt-action-analyze-ccpp-local.1.gz
abrt-action-analyze-core.1.gz
abrt-action-analyze-java.1.gz
abrt-action-analyze-oops.1.gz
abrt-action-analyze-python.1.gz
abrt-action-analyze-vmcore.1.gz
abrt-action-analyze-vulnerability.1.gz
abrt-action-analyze-xorg.1.gz
abrt-action-check-oops-for-hw-error.1.gz
abrt-action-find-bodhi-update.1.gz
abrt-action-generate-backtrace.1.gz
abrt-action-generate-core-backtrace.1.gz
abrt-action-install-debuginfo.1.gz
```

Рис. 2.3: Каталог

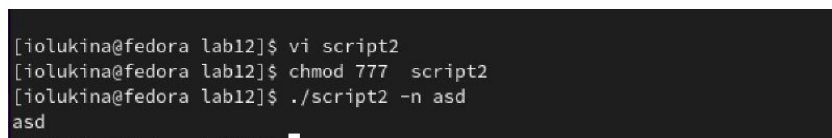


```
iolukina@fedora:~/lab12 — /usr/libexec/vi script2
command=""

while getopts :n: opt
do
case $opt in
n) command="$OPTARG";;
esac
done

echo "$command"
```

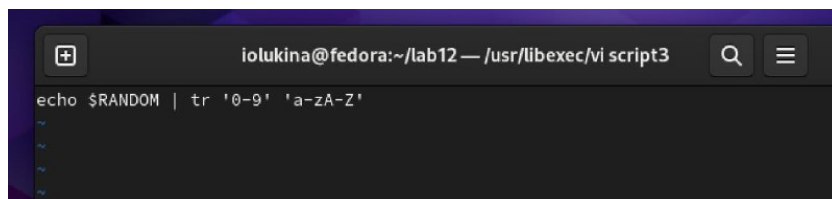
Рис. 2.4: Файл



```
[iolukina@fedora lab12]$ vi script2
[iolukina@fedora lab12]$ chmod 777 script2
[iolukina@fedora lab12]$ ./script2 -n asd
asd
```

Рис. 2.5: Работа файла

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



```
iolukina@fedora:~/lab12 — /usr/libexec/vi script3
echo $RANDOM | tr '0-9' 'a-zA-Z'
```

Рис. 2.6: Код


```
ciafd  
[iolukina@fedora lab12]$ ./script3  
bbiaj  
[iolukina@fedora lab12]$ ./script3  
igfh  
[iolukina@fedora lab12]$ ./script3  
befab  
[iolukina@fedora lab12]$ █
```

Рис. 2.7: Работа файла

3 Выводы

В ходе выполнения работы я приобрела практические навыки программирования в системе Линукс

Список литературы