

Программирование в командном процессоре ОС UNIX. Расширенное программирование

Лабораторная работа №12

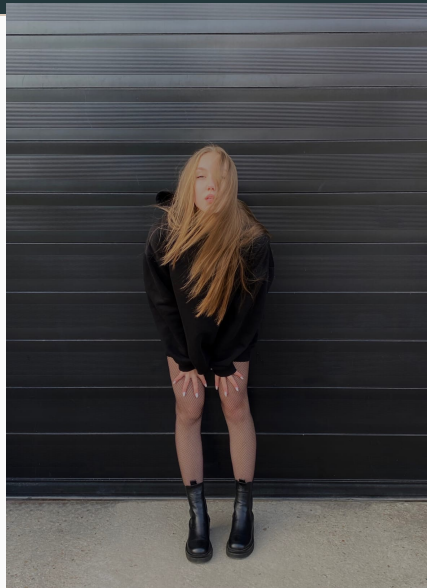
Миронова М. В.

29 апреля 2023

Российский университет дружбы народов, Москва, Россия

Информация

- Миронова М. В.
- студент 1 курса, группа НММбд-03-22
- Российский университет дружбы народов



Вводная часть

- Командный процессор ОС UNIX
- Командные файлы


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

- Ознакомиться с теоретическим материалом.
- Выполнить упражнения.
- Ответить на контрольные вопросы.

Выполнение лабораторной работы №12

Первая программа

```
[mvmironova@fedora ~]$ touch lab12_1.sh  
[mvmironova@fedora ~]$
```

```
Открыть ▾  • lab12_1.sh  
~  
  
#!/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  
t1=$1  
t2=$2  
command=$3  
while true  
do  
    if [ "$command" == "Выход" ]  
    then  
        break  
    fi  
done
```

```
[mvmironova@fedora ~]$ chmod +x lab12_1.sh
```

Первая программа доработка

```
[mvmironova@fedora ~]$ ./lab12_1.sh 3 7
```

```
Ожидайте  
Ожидайте  
Ожидайте  
Выполнение  
Выполнение  
Выполнение  
Выполнение  
Выполнение  
Выполнение  
Выполнение
```

Открыть ▾ +

lab12_1.sh

~/

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

Вторая программа

```
[mvmironova@fedora ~]$ ./lab12_1.sh 4 3 2
Ожидайте
Ожидайте
Ожидайте
Ожидайте
Выполнение
Выполнение
Выполнение
Следующее действие
Следующее действие
```

```
[mvmironova@fedora ~]$ cd /usr/share/man/man1
[mvmironova@fedora man1]$ ls
i.1.gz
'.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
```

Вторая программа

```
[mvmironova@fedora ~]$ touch lab12_2.sh  
[mvmironova@fedora ~]$
```

```
Открыть ▾  lab12_2.sh  
~/  
  
#!/bin/bash  
c=$1  
if [ -f /usr/share/man/man1/${c}.1.gz ]  
then  
    gunzip -c /usr/share/man/man1/${c}.1.gz | less  
else  
    echo "Справка по данной команде нет"  
fi
```

```
[mvmironova@fedora ~]$ chmod +x lab12_2.sh
```

Третья программа

```
[mvmironova@fedora ~]$ ./lab12_2.sh ls  
[mvmironova@fedora ~]$ ./lab12_2.sh cd
```

```
mvmironova@fedora:~ -- /bin/bash ./lab12_2.sh ls  
." DO NOT MODIFY THIS FILE! It was generated by help2man 1.48.5.  
.TH LS "1" "March 2022" "GNU coreutils 9.0" "User Commands"  
..SH NAME  
ls \- list directory contents  
..SH SYNOPSIS  
..B ls  
[ \FI,OPTION\ \FR]... [ \FI,FILE\ \FR]...  
..SH DESCRIPTION  
.\" Add any additional description here  
..PP  
List information about the FILES (the current directory by default).  
Sort entries alphabetically if none of \FB\ -ctuvSUX\FR nor \FB\ -\sort\FR is sp  
ecified.  
..PP  
Mandatory arguments to long options are mandatory for short options too.  
..TP  
\FB\ -a\FR, \FB\ -\all\FR  
do not ignore entries starting with .  
..TP  
\FB\ -A\FR, \FB\ -\almost-\all\FR  
do not list implied . and ..  
..TP  
\FB\ -\author\FR
```

```
mvmironova@fedora:~ -- /bin/bash ./lab12_2.sh cd  
..so man1/builtins.1
```

Ответы на контрольные вопросы

```
[mvmironova@fedora ~]$ touch lab12_3.sh
```

Открыть ▾



• lab12_3.sh

~/

```
#!/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
```

Результаты

В ходе выполнения лабораторной работы были изучены основы программирования в оболочке ОС UNIX. Научились писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.