

Программирование в командном процессоре ОС UNIX.

Ветвления и циклы

Ekaterina D. Krutova¹

NEC-2022, 31 Feb – 31 Feb, 2022 Moscow, Russia

¹RUDN University, Moscow, Russian Federation

Цель работы

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

Ход работы

Выполнение рис. 1-5

Задание 1 (1)

```
#!/bin/bash

while getopts :i:o:p:cn opt
do
    case "${opt}" in
        i)input=${OPTARG};;
        o)output=${OPTARG};;
        p)mask=${OPTARG};;
        c)c=true;;
        n)n=true;;
    esac
done

if [ $c ]
then
    if [ $n ]
    then
        grep -n -i $mask $input > $output
        exit 0
    else
        grep -i $mask $input > $output
        exit 0
    fi
else
    grep -n $mask $input > $output
    exit 0
fi

#!/bin/bash

while getopts "i:o:p:Cn" opt; do
    case $opt in
        i)
            input_file=$OPTARG
            ;;
        o)
            output_file=$OPTARG
            ;;
    esac
done
```

Figure 1: Текст командного файла к заданию 1 (1)

Задание 1 (2)

```
output_file=$OPTARG
;;
p)
pattern=$OPTARG
;;
C)
case_insensitive=1
;;
n)
print_line_numbers=1
;;
\?)
echo "Invalid option: -$OPTARG" >&2
exit 1
;;
:)
echo "Option -$OPTARG requires an argument." >&2
exit 1
;;
esac
done

if [ -z "$input_file" ] || [ -z "$output_file" ] || [ -z "$pattern" ]; then
    echo "Usage: $0 -i <input_file> -o <output_file> -p <pattern> [-C] [-n]"
    exit 1
fi

grep ${if [ -n "$case_insensitive" ]; then echo "-i"; fi} ${if [ -n "$print_line_numbers" ]; then
printf %s "-n"; fi} "$pattern" "$input_file" > "$output_file"
```

Figure 2: Текст командного файла к заданию 1 (2)

Задание 2

```
#!/bin/bash

gcc f1.c -o f1
./f1
code=$?
case $code in
    0) echo "chislo <0";;
    1) echo "chislo >0";;
    2) echo "chislo =0";;
esac
```

Figure 3: Текст командного файла к заданию 2

Задание 3

```
#!/bin/bash

opt=$1;
form=$2;
num=$3;
function files(){
    for ((i=1; i<=$num; i++)) do
        file=$(echo $form | tr '#' "$i")
        if [ $opt == "-r" ]
        then
            rm -f $file
        elif [ $opt == "-c" ]
        then
            touch $file
        fi
    done
}
files
```

Figure 4: Текст командного файла к заданию 3

Задание 4

```
#!/bin/bash

rm -f archived.tar

for file in $(find $1 -type f -mtime -7); do
    echo $file
    if [ -f archived.tar ]; then
        tar rf archived.tar "$file" 2> /dev/null
    else
        tar cf archived.tar "$file" 2> /dev/null
    fi
done
```

Figure 5: Текст командного файла к заданию 4

Вывод

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