Отчёт по лабораторной работе №11

Программирование в командной процессоре ОС UNIX. Ветвления и циклы

Никифоров Георгий Сергеевич

Содержание

## 0.1 Цель работы

Цель работы — изучить основы программирования в командной оболочке OS Unix.

# 1 Выполнение лабораторной работы

Необходимо выполнить задания:

1. Используя команды getopts grep, написать командный файл, который анализирует командную строку с ключами: – -iinputfile — прочитать данные из указанного файла; – -ooutputfile — вывести данные в указанный файл; – -pшаблон — указать шаблон для поиска; – -C — различать большие и малые буквы; – -n — выдавать номера строк. а затем ищет в указанном файле нужные строки, определяемые ключом -p.

Данная задача была решена в файле stepa.sh:

#!/bin/bash  
# Unpacking parameters  
while getopts i:o:p:Cn flag  
do  
 case $flag in  
 i) inputFile=$OPTARG;;  
 o) outputFile=$OPTARG;;  
 p) pattern=$OPTARG;;  
 C) C='--color=always'; echo Flag -$flag will switch color output on;;  
 n) n=n;;  
 \*) echo Illegal option $flag used!;;  
 esac  
done  
touch $outputFile  
grep $C -${n}e $pattern $inputFile > $outputFile

1. Написать на языке C программу, которая вводит число и определяет, является ли оно больше нуля, меньше нуля или равно нулю. Затем программа завершается с помощью функции exit(n), передавая информацию в о коде завершения в оболочку. Командный файл должен вызывать эту программу и, проанализировав с помощью команды $?, выдать сообщение о том, какое число было введено.

baton.c:

#include <stdlib.h>  
#include <stdio.h>  
int main(int argc, char\* argv[])  
{  
 char\* govno = argv[1];  
 int i = atoi(govno);  
 if (i < 0)  
 { exit(1); }  
 else if (i > 0)  
 { exit(2); }  
 exit(0);  
}

fedya.sh:

#!/bin/bash  
if (($# < 1))  
then  
 echo fedya needs to eat more than one argument \  
 otherwise he will not work.  
 exit  
fi  
./baton $1  
exit=$?  
case $exit in  
0) type=zeroish;;  
1) type=negative;;  
2) type=positive;;  
\*) type=stupid;;  
esac  
echo Exit code is $exit so it was $type number.

1. Написать командный файл, создающий указанное число файлов, пронумерованных последовательно от 1 до N (например 1.tmp, 2.tmp, 3.tmp,4.tmp и т.д.). Число файлов, которые необходимо создать, передаётся в аргументы командной строки. Этот же командный файл должен уметь удалять все созданные им файлы (если они существуют).

smekhail.sh:

#!/bin/bash  
ext=.tmp  
function CreateFiles()  
{  
 i=1  
 while ((i <= $1))  
 do  
 echo Creating $i$ext  
 touch $i$ext  
 let i++  
 done  
}  
function RemoveFiles()  
{  
 i=1  
 while ((i<= $1))  
 do  
 echo Removing $i$ext  
 rm $i$ext  
 let i++  
 done  
}  
if (($# < 1))  
then  
 echo Needed at least one parameter. Terminate  
 exit  
fi  
min=1  
if (($1 < 1))  
then  
 echo N is less than $min. Assuming N = $min.  
 N=$min  
else  
 N=$1  
fi  
CreateFiles $N  
echo -------------------------  
RemoveFiles $N

1. Написать командный файл, который с помощью команды tar запаковывает в архив все файлы в указанной директории. Модифицировать его так, чтобы запаковывались только те файлы, которые были изменены менее недели тому назад (использовать команду find).

potom.sh:

#!/bin/bash  
if (($# < 1))  
then  
 target=.  
else  
 target=$1  
fi  
outputFile=$(pwd)\/archive.tar  
tar -cf $outputFile $(find $target -maxdepth 1 -atime -7 -type f)  
if (($? == 0))  
then  
 echo Successfully archived following files into $target:  
 tar -tf $outputFile  
fi

## 1.1 Ответы на контрольные вопросы

1. Каково предназначение команды getopts?

*Ответ*: команда анализирует аргументы, переданные скрипту.

1. Какое отношение метасимволы имеют к генерации имён файлов?

*Ответ*: метасимволы позволяют создавать файлы, используя шаблоны.

1. Какие операторы управления действиями вы знаете?

*Ответ*: никакие.

1. Какие операторы используются для прерывания цикла?

*Ответ*: break, continue.

1. Для чего нужны команды false и true?

*Ответ*: false всегда возвращает код завершения, не равный нулю (т. е. ложь). Команда true выполняет обратное действие.

1. Что означает строка if test -f man$s/$i.$s, встреченная в командном файле?

*Ответ*: це проверка условия на наличие файла с шаблоном имени if test -f man$s/$i.$s.

1. Объясните различия между конструкциями while и until.

*Ответ*: при замене в операторе цикла while служебного слова while на until условие, при выполнении которого осуществляется выход из цикла, меняется на противоположное. В остальном оператор цикла while и оператор цикла until идентичны..

## 1.2 Заключение

В ходе выполнения лабораторной работы были изучены основы программирования в командной оболочке OS Unix. Цель работы была достигнута.