

Отчет по лабораторной работе № 21 по курсу «Практикум на ЭВМ»

Студент группы М8О-105Б-21 Минеева Светлана Алексеевна, № по списку 14

Контакты e-mail: svetlana.mineewa2003@yandex.ru

Работа выполнена: «22» марта 2022 г.

Преподаватель: Вячеслав Константинович Титов каф. 805

Отчет сдан «22» марта 2022 г., итоговая оценка _____

Подпись преподавателя _____

1. **Тема:** Программирование на интерпретируемых командных языках.
2. **Цель работы:** Составить программу выполнения заданных действий над файлами на одном из интерпретируемых командных языках ОС UNIX.
3. **Задание (вариант № 14):** Удаление файлов с указанным префиксом, длина которых находится в указанном диапазоне.
4. **Оборудование (лабораторное):**
ЭВМ _____, процессор _____, имя узла сети _____ с ОП _____ Мб,
НМД _____ Мб. Терминал _____ адрес _____. Принтер _____
Другие устройства _____

Оборудование ПЭВМ студента, если использовалось:

Процессор 2,9 GHz 2-ядерный процессор Intel Core i5 с ОП 8 Гб, НМД 500 Гб. Монитор 13,3-дюймовый (2560 x 1600).

Другие устройства _____

5. **Программное обеспечение (лабораторное):**
Операционная система семейства _____, наименование _____ версия _____
интерпретатор команд _____ версия _____
Система программирования _____ версия _____
Редактор текстов _____ версия _____
Утилиты операционной системы _____
Прикладные системы и программы _____
Местонахождение и имена файлов программ и данных _____

Программное обеспечение ЭВМ студента, если использовалось:

Операционная система семейства UNIX, наименование Terminal версия 2.10

интерпретатор команд bash версия 3.2.

Система программирования _____ версия _____

Редактор текстов emacs версия 27.2-2

Утилиты операционной системы cat, ls, cd, chmod и другие

Прикладные системы и программы _____

Местонахождение и имена файлов программ и данных на домашнем компьютере _____

6. Идея, метод, алгоритм решения задачи (в формах: словесной, псевдокода, графической [блок-схема, диаграмма, рисунок, таблица] или формальные спецификации с пред- и постусловиями)

Алгоритм работы:

Проверяем количество введенных аргументов и при неправильном вводе выводим на экран пример верного ввода данных. Проходя циклом по файлам с заданным префиксом, присваиваем переменной значение длины этого файла. Сравниваем значение этой переменной с заданными значениями начала и конца диапазона. В случае выполнения этих условий удаляем файл. Если при запуске программы использовался флаг -i, то перед удалением файла выводим на экран вопрос об удалении с вариантом выбора – да или нет, дополнительно в условия добавляется проверка, что на вопрос был дан ответ «да».

7. Сценарий выполнения работы [план работы, первоначальный текст программы в черновике (можно на отдельном листе) и тесты либо соображения по тестированию].

```
#!/bin/bash
```

```
if [ $# = 0 ]
then echo "I've done nothing"; exit;fi
if [[ $1 == ? ]] || [[ $2 == " " ]] || [[ $3 == " " ]]
then echo "Use: ./script [prefix] [range start] [range end]";exit;fi
if [[ $1 == '-i' ]]
then i=1; shift
else i=0;fi
echo "i=$i $1=$1 \#$=# $2=$2 $3=$3"
k2=$2; k3=$3
if [ $# = 0 ];then
for file in *
do
echo "do delete file $file ? (y,n)"
read a
size=$(wc -c $file | awk '{print $1}')
if [[ $a == 'y' ]] && [[ $size -ge $k2 ]] && [[ $size -le $k3 ]];then
rm $file;fi
done
else
for file in $1*;do
size=$(wc -c $file | awk '{print $1}')
if [ $i = 1 ]
then
echo "do delete file $file ? (y,n)"
read a
if [[ $a == 'y' ]] && [[ $size -ge $k2 ]] && [[ $size -le $k3 ]];then
rm $file;fi
else
echo "size=$size k2=$k2 k3=$k3"
if [[ $size -ge $k2 ]] && [[ $size -le $k3 ]];then
rm $file
echo $file deleted; fi
fi
done
fi
```

Тесты:

- 1) Файл file1 с длиной 13: Hello world!
Файл file2 с длиной 8: My home
Префикс: fi

- Диапазон: [6,10]
Итог: удаление файла file2
- 2) Файл text1 с длиной 20: This is my homework
Файл text2 с длиной 10: black cat
Префикс: te
Диапазон: [9,12]
Итог: удаление файла text2
- 3) Префикс: le
Диапазон: [4,7]
Итог: ни один файл не удалён

Пункты 1-7 отчета составляются строго до начала лабораторной работы.

8. Распечатка протокола (подклеить листинг окончательного варианта программы с тестовыми примерами, подписанный преподавателем).

Last login: Tue Mar 22 16:15:32 on ttys000

The default interactive shell is now zsh.
To update your account to use zsh, please run `chsh -s /bin/zsh`.
For more details, please visit <https://support.apple.com/kb/HT208050>.
MacBook-Pro-MacBook:~ macbookpro\$ cat zag.txt

```
* Минеева Светлана Алексеевна *
*      М80-105Б-21      *
*   Лабораторная работа №21   *
```

MacBook-Pro-MacBook:~ macbookpro\$ cd lab21

MacBook-Pro-MacBook:lab21 macbookpro\$ ls

file1 file2 script text1 text2

MacBook-Pro-MacBook:lab21 macbookpro\$ cat script

#!/bin/bash

```
if [ $# = 0 ]
then echo "I've done nothing"; exit;fi
if [[ $1 == ? ]] || [[ $2 == " ]] || [[ $3 == " ]]
then echo "Use: ./script [prefix] [range start] [range end]";exit;fi
if [[ $1 == '-i' ]]
then i=1; shift
else i=0;fi
echo "i=$i \ $1=$1 \ $#=$# \ $2=$2 \ $3=$3"
k2=$2; k3=$3
if [ $# = 0 ];then
for file in *
do
echo "do delete file $file ? (y,n)"
read a
size=$(wc -c $file | awk '{print $1}')
if [[ $a == 'y' ]] && [[ $size -ge $k2 ]] && [[ $size -le $k3 ]];then
rm $file;fi
done
else
for file in $1*;do
size=$(wc -c $file | awk '{print $1}')
if [ $i = 1 ]
then
echo "do delete file $file ? (y,n)"
read a
if [[ $a == 'y' ]] && [[ $size -ge $k2 ]] && [[ $size -le $k3 ]];then
rm $file;fi
else
echo "size=$size k2=$k2 k3=$k3"
if [[ $size -ge $k2 ]] && [[ $size -le $k3 ]];then
rm $file
echo $file deleted; fi
fi
done
fi
```

MacBook-Pro-MacBook:lab21 macbookpro\$ cat file1

Hello world!

MacBook-Pro-MacBook:lab21 macbookpro\$ wc -c file1

13 file1

MacBook-Pro-MacBook:lab21 macbookpro\$ cat file2

My home

MacBook-Pro-MacBook:lab21 macbookpro\$ wc -c file2

```
8 file2
MacBook-Pro-MacBook:lab21 macbookpro$ cat text1
This is my homework
MacBook-Pro-MacBook:lab21 macbookpro$ wc -c text1
20 text1
MacBook-Pro-MacBook:lab21 macbookpro$ cat text2
black cat
MacBook-Pro-MacBook:lab21 macbookpro$ wc -c text2
10 text2
MacBook-Pro-MacBook:lab21 macbookpro$ chmod +x ./script
MacBook-Pro-MacBook:lab21 macbookpro$ ./script
I've done nothing
MacBook-Pro-MacBook:lab21 macbookpro$ ./script 3 4 5
Use: ./script [prefix] [range start] [range end]
MacBook-Pro-MacBook:lab21 macbookpro$ ./script fi 6 10
i=0 $1=fi $#=3 $2=6 $3=10
size=13 k2=6 k3=10
size=8 k2=6 k3=10
file2 deleted
MacBook-Pro-MacBook:lab21 macbookpro$ ls
file1  script  text1  text2
MacBook-Pro-MacBook:lab21 macbookpro$ ./script -i te 9 12
i=1 $1=te $#=3 $2=9 $3=12
do delete file text1 ? (y,n)
y
do delete file text2 ? (y,n)
y
MacBook-Pro-MacBook:lab21 macbookpro$ ls
file1  script  text1
MacBook-Pro-MacBook:lab21 macbookpro$ ./script le 4 7
i=0 $1=le $#=3 $2=4 $3=7
wc: le*: open: No such file or directory
size= k2=4 k3=7
MacBook-Pro-MacBook:lab21 macbookpro$ ls
file1  script  text1
MacBook-Pro-MacBook:lab21 macbookpro$
```

9. Дневник отладки должен содержать дату и время сеансов отладки и основные события (ошибки в сценарии и программе, нестандартные ситуации) и краткие комментарии к ним. В дневнике отладки приводятся сведения об использовании других ЭВМ, существенном участии преподавателя и других лиц в написании и отладке программы.

№	Лаб. или дом.	Дата	Время	Событие	Действие по исправлению	Примечание
1	Дом.	22.03.22	14:00	Ошибка в работе оператора if	Ошибка исправлена; более подробно изучено введение условий к оператору if	Нужно заключать условия в двойные квадратные скобки, чтобы не возникало ошибки - «unary operator expected»

10. Замечания автора по существу работы

Эта лабораторная работа очень полезна, она отлично развивает мышление и учит программированию на интерпретируемом командном языке.

11. Выводы

Я составила программу выполнения заданных действий над файлами на интерпретируемом командном языке ОС UNIX Bash.

Недочёты при выполнении задания могут быть устранены следующим образом: больше практиковаться в написании программ на интерпретируемом командном языке Bash.

Подпись студента Минеева С.А