

Отчет по лабораторной работе № 21 по курсу “Языки и методы программирования”

Студент группы М8О-111Б-21, Суляева Алина Игоревна
№ по списку 22

Контакты www, e-mail:
alexander.iatsenko@gmail.com

Работа выполнена: « » 201 ___ г.

Преподаватель: Никулин Сергей Петрович Каф.
806 _____

Входной контроль знаний с оценкой _____

Отчет сдан « » 201 __ г., итоговая оценка _____

Подпись преподавателя _____

1. Тема: программирование на интерпретируемых командных языках (bash)

2. Цель работы: составить программу выполнения заданных действий над файлами на одном из интерпретируемых командных языков ОС UNIX

3. Задание: Удаление всех файлов с указанным суффиксом при наличии файлов с теми же именами, но с другим заданным суффиксом. Дополнительные условия: 1. Если параметры опущены, то их значения должны быть запрошены у пользователя; 3. Если указан параметр “?”, то выводится подсказка (спецификация программы) (задание 7)

4. Оборудование:

Оборудование ПЭВМ студента:

Процессор: AMD Ryzen 5 4600H, с ОП 6 Мб (виртуальная машина), НМД 25600 Мб. Монитор: Huawei 16,1 IPS 1920×1080, 137 РР.

5. Программное обеспечение ЭВМ студента, если использовалось:

Операционная система UNIX fedora

Прикладные системы и программы: терминал ОС UNIX, текстовый редактор emacs

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

Цель программы – удаление всех файлов с заданным суффиксом при наличии хотя бы одного совпадающего имени с файлами с другими суффиксами.

Для начала необходимо создать такие файлы. Для этого был написана специальная программа. Она генерирует случайный суффикс через /dev/urandom, затем создаёт случайное число от 0 до 99, потом убирает ведущие нули (nums). remaining length – оставшаяся длина после суффикса. Далее она заходит в цикл и генерирует случайное число от 1 до 2, если число 1, то программа создаёт файл со сгенерированным суффиксом посредством утилиты dd, дальше создаёт второе случайное число от 1 до 3, и если оно равно 1, то создаёт ещё один файл с таким же названием посредством утилиты dd, но с другим суффиксом (вероятность 33%). Если первое случайное равно 2, тогда создаёт случайный файл со случайным названием и случайным суффиксом посредством утилиты

dd. В конце выводится суффикс, который был сгенерирован. На выходе имеем множество сгенерированных новых файлов в текущей папке.

Основная программа выводит спецификацию при задании флага “?” (дополнительное условие) и через getopt можно ввести переменную suf, если ничего не было введено (-z \$suf), то программа будет запрашивать суффикс, пока пользователь его не введёт (третье дополнительное условие).

Далее программа добавит точку к суффиксу (suf="echo .'\$suf"), ищет все файлы с данным суффиксом посредством утилиты find и через xargs -l (-l служит для построчного ввода в файл) записывает все такие файлы в файл temp посредством утилиты echo (find -name "*\$suf" | xargs -l echo > temp). Для наглядности программы работы пользователю список выводится через temp.

Далее программа удаляет ./ в начале и \$suf в конце в созданном файле temp посредством потокового текстового редактора sed, используя регулярные выражения (cat temp | sed "s/..//g" | sed "s/\$suf//g" | xargs -l echo > temp): s внутри sed меняет символ в формате s/символ/другойсимвол/. “.” – любой случайный символ в начале, гарантированно “./”.

Далее создаётся список файлов other с суффиксами, отличными (посредством знака “!”) от \$suf (find ! -name "*\$suf" ! -name "temp" ! -name "." ! -name "other" ! -name "all" | xargs -l echo > other). Также исключается созданный файл temp, “.” в списке файлов, “other” (программа сначала создаёт файл, а потом в него записывает). Выводится other.

Наконец, в цикле

```
while read x
    do
        cnt=`cat other | grep "^.*/${x}\b" | wc -l`
        if [[ cnt -ne 0 ]]
            then
                find -name "*$suf" -delete
                break
        fi
    done < temp

echo "Чистка мусора: удаление списков temp и other"
```

Читается переменная x – построчный вывод имён файлов из temp, вводится переменная cnt для количества путём сравнения с 0 посредством утилиты wc -l и чтения файлов из списка other, используя grep "^.*/\${x}\b". ^ указывает на начало имени файла, ./ – первые два символа, \${x} – искомое имя, \b – граница слова. Если таких файлов нет, то удаление не произойдёт. Иначе (cnt >= 1) программа ищет все файлы с исходным суффиксом и удаляет их. Далее происходит чистка мусор путём удаления temp и other (rm temp other).

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

Код программы

```

while read x
do
    cnt=`cat other | grep "^.*/${x}\b" | wc -l`
    if [[ cnt -ne 0 ]]
        then
            find -name "*$suf" -delete
            break
    fi
done < temp

echo "Чистка мусора: удаление списков temp и other"

```

```

#!/bin/bash
# Удаление всех файлов с указанным суффиксом при наличии файлов с теми же именами,
# но с другим заданным суффиксом
# bash -x
if [[ $1 == "?" ]]; then
    echo -ne "Программа получает на вход суффикс, далее она создаёт список фай-
    лов temp с указанным суффиксом.\n"
    echo -ne "Список temp обрабатывается так, что в нём остаются только имена файлов. Да-
    лее создаётся список other из файлов \n"
    echo -ne "с отличными суффиксами от указанного. После чего программа захо-
    дит в цикл по списку temp и при наличии хотя бы \n"
    echo -ne "одного совпадающего имени у temp и other удаляет ВСЕ файлы с исходным суффик-
    ком.\n"
    exit 0
fi

# Задание суффикса через флаг для первого условия
while getopts ":s:" opt;
do
    case $opt in
        s)
            suf="$OPTARG";;
        *)
            echo "Вы ввели неизвестный флаг. Введите \"./lab21.sh ?\" для справки";
    done

# Проверка на пустой символ
while [[ -z $suf ]]
do
    echo "Введите суффикс: "
    read suf
done

# Добавление точки к суффиксу
suf=`echo .`$suf`

echo "Создание списка файлов temp с указанным суффиксом $suf: "
find -name "*$suf" | xargs -l echo > temp
cat temp

echo "Удаление ./ в начале и $suf в конце для temp (необходимо для сравнения имён): "
cat temp | sed "s///" | sed "s/$suf//" | xargs -l echo > temp
cat temp

echo "Создание списка файлов other с суффиксами, отличными от $suf"
find ! -name "*$suf" ! -name "temp" ! -name "." ! -name "other" | xargs -l echo > other
cat other

```

```

echo "Удаление всех файлов с $suf при наличии хотя бы одного совпадающего имени"
#читывает первую строку как x в temp, ищет такое же имя через grep, затем уда-
ляют ВСЕ файлы с указанным суффиксом
#иначе не делает ничего
while read x
do
    cnt=`cat other | grep "^.*/${x}\b" | wc -l`
    if [[ cnt -ne 0 ]]
        then
            find -name "*$suf" -delete
            break
    fi
done < temp

echo "Чистка мусора: удаление списков temp и other"
rm temp other

```

Код программы для создания файлов для тестов

```
#!/bin/bash

counter="0"

suffix=$( cat /dev/urandom | tr -dc 'a-zA-Z0-9' | fold -w $(cat /dev/urandom | tr -dc '1-9' | fold -w 32 | head -c 1) | head -n 1 )
nums=$( cat /dev/urandom | tr -dc '0-9' | fold -w 32 | head -c 2 )
let nums="10#${nums}"

let remainingLength=8-${#suffix}

while [[ counter -lt nums ]]; do
    random=$( cat /dev/urandom | tr -dc '1-2' | fold -w 32 | head -c 1)
    if [[ random -eq 1 ]]; then
        cur_prefix=$( cat /dev/urandom | tr -dc 'a-zA-Z0-9' | fold -w 8 | head -c $remainingLength )
        dd if=/dev/urandom of="${PWD}/${cur_prefix}${suffix}" bs=1 count=1
        random2=$( cat /dev/urandom | tr -dc '1-3' | fold -w 32 | head -c 1)
        if [[ random2 -eq 1 ]]; then
            dd if=/dev/urandom of="${PWD}/${cur_prefix}$(echo .)$(cat /dev/urandom | tr -dc 'a-zA-Z0-9' | fold -w 8 | head -c 3)" bs=1 count=1
        fi
        elif [[ random -eq 2 ]]; then
            dd if=/dev/urandom of="${PWD}/$( cat /dev/urandom | tr -dc 'a-zA-Z0-9' | fold -w 8 | head -n 1 )$(echo .)$(cat /dev/urandom | tr -dc 'a-zA-Z0-9' | fold -w 8 | head -c 3 )" bs=1 count=1
        fi
        let counter=$counter+1
done

echo -ne "Суффикс = \"$suffix\"\n"
```

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

Допущен к выполнению работы. Подпись преподавателя

8. Распечатка протокола (подклейте листинг окончательного варианта программы с тестовыми примерами, подписаный преподавателем)

```
ucvim@UCvim-PC:/mnt/c/Users/UCvim/Google Drive/MAI/FI & VS/Nikulin/labs/l21$ cat l21
```

Суляева Алинв, М80-111Б-21
Лабораторная работа №21

```
ucvim@UCvim-PC:/mnt/c/Users/UCvim/Google Drive/MAI/FI & VS/Nikulin/labs/l21$ cd test
ucvim@UCvim-PC:/mnt/c/Users/UCvim/Google Drive/MAI/FI & VS/Nikulin/labs/l21/test$ cat ..//tester.sh #!/bin/bash
counter="0"
suffix=$( cat /dev/urandom | tr -dc 'a-zA-Z0-9' | fold -w $(cat /dev/urandom | tr -dc '1-9' | fold -w 32 | head -c 1) | head -n 1 )
nums=$( cat /dev/urandom | tr -dc '0-9' | fold -w 32 | head -c 2 ) let nums="10#${nums}"
let remainingLength=8-$suffix
while [[ counter -lt nums ]]; do
random=$( cat /dev/urandom | tr -dc '1-2' | fold -w 32 | head -c 1 ) if [[ random -eq 1 ]]; then
cur_prefix=$( cat /dev/urandom | tr -dc 'a-zA-Z0-9' | fold -w 8 | head -c $remainingLength ) dd if=/dev/urandom of="$PWD/${cur_prefix}$(echo .)$suffix" bs=1 count=1 random2=$( cat /dev/urandom | tr -dc '1-3' | fold -w 32 | head -c 1 )
if [[ random2 -eq 1 ]]; then
dd if=/dev/urandom of="$PWD/${cur_prefix}$(echo .)(cat /dev/urandom | tr -dc 'a-zA-Z0-9' | fold -w 8 | head -c 3)" bs=1 count=1
fi
elif [[ random -eq 2 ]]; then
dd if=/dev/urandom of="$PWD/$( cat /dev/urandom | tr -dc 'a-zA-Z0-9' | fold -w 8 | head -n 1 )$(echo .)$(cat /dev/urandom | tr -dc 'a-zA-Z0-9' | fold -w 8| head -c 3 )" bs=1 count=1
fi
let counter=$counter+1 done
echo -ne "Суффикс = \"$suffix\"\n"
ucvim@UCvim-PC:/mnt/c/Users/UCvim/Google Drive/MAI/FI & VS/Nikulin/labs/l21/test$ cat ..//l21.sh #!/bin/bash
# Удаление всех файлов с указанным суффиксом при наличии файлов с теми же именами,
# но с другим заданным суффиксом
# bash -x
if [[ $1 == "?" ]]; then
echo -ne "Программа получает на вход суффикс, далее она создаёт список файлов temp с указанным суффиксом.\n"
echo -ne "Список temp обрабатывается так, что в нём остаются только имена файлов. Далее создаётся список other из файлов \n"
echo -ne "с отличными суффиксами от указанного. После чего программа заходит в цикл по списку temp и при наличии хотя бы \n"
echo -ne "одного совпадающего имени у temp и other удаляет ВСЕ файлы с исходным суффиксом.\n"
exit 0 fi
# Задание суффикса через флаг для первого условия while getopts ":s:" opt;
do
case $opt in s)
*)
echo "Вы ввели неизвестный флаг. Введите \"./lab21.sh ?\" для справки"; exit 1;;
esac done
```

```
# Проверка на пустой символ while [[ -z $suf ]]
do
echo "Введите суффикс: " read suf
done
suf="$OPTARG";;
# Добавление точки к суффиксу suf=`echo .`$suf"
echo "Создание списка файлов temp с указанным суффиксом $suf: " find -name "*$suf" | xargs -l echo > temp
cat temp
echo "Удаление ./ в начале и $suf в конце для temp (необходимо для сравнения имён): " cat temp | sed "s/..//"
| sed "s/$suf//"
| xargs -l echo > temp
cat temp
echo "Создание списка файлов other с суффиксами, отличными от $suf"
find ! -name "*$suf" ! -name "temp" ! -name "." ! -name "other" ! -name "all" | xargs -l echo > other
cat other
echo "Удаление всех файлов с $suf при наличии хотя бы одного совпадающего имени"
#читывает первую строку как x в temp, ищет такое же имя через grep, затем удаляет ВСЕ файлы с
указанным суффиксом
#иначе не делает ничего while read x
do
cnt=`cat other | grep "^.*/${x}\b" | wc -l` if [[cnt-ne0]]
then
find -name "*$suf" -delete break
fi done < temp
echo "Чистка мусора: удаление списков temp и other"
rm temp other
ucvim@UCvim-PC:/mnt/c/Users/UCvim/Google Drive/MAI/FI & VS/Nikulin/labs/l21/test$ ./tester.sh 1+0
records in
1+0 records out
1 byte copied, 0.0010038 s, 1.0 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0008023 s, 1.2 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0010558 s, 0.9 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0008181 s, 1.2 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0007393 s, 1.4 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0007718 s, 1.3 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0007195 s, 1.4 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0008393 s, 1.2 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.000854 s, 1.2 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0009307 s, 1.1 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0008947 s, 1.1 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0009091 s, 1.1 kB/s 1+0 records in
```

1+0 records out
1 byte copied, 0.0010854 s, 0.9 kB/s
1+0 records in
1+0 records out
1 byte copied, 0.0008867 s, 1.1 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0011241 s, 0.9 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0008438 s, 1.2 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.000884 s, 1.1 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0010412 s, 1.0 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0010136 s, 1.0 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0007746 s, 1.3 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0007347 s, 1.4 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0008275 s, 1.2 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0007383 s, 1.4 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0013956 s, 0.7 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.000776 s, 1.3 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0007537 s, 1.3 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0007404 s, 1.4 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0007677 s, 1.3 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0007816 s, 1.3 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0009531 s, 1.0 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0007219 s, 1.4 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0008115 s, 1.2 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0007567 s, 1.3 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0007386 s, 1.4 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0009586 s, 1.0 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0007981 s, 1.3 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0010333 s, 1.0 kB/s 1+0 records in
1+0 records out

1 byte copied, 0.0008097 s, 1.2 kB/s 1+0 records in
1+0 records out

1 byte copied, 0.001061 s, 0.9 kB/s 1+0 records in
1+0 records out

1 byte copied, 0.000888 s, 1.1 kB/s 1+0 records in
1+0 records out

1 byte copied, 0.0009881 s, 1.0 kB/s 1+0 records in
1+0 records out

1 byte copied, 0.0007658 s, 1.3 kB/s 1+0 records in
1+0 records out

1 byte copied, 0.0008108 s, 1.2 kB/s 1+0 records in
1+0 records out

1 byte copied, 0.0009135 s, 1.1 kB/s 1+0 records in
1+0 records out

1 byte copied, 0.0008875 s, 1.1 kB/s 1+0 records in
1+0 records out

1 byte copied, 0.0007437 s, 1.3 kB/s 1+0 records in
1+0 records out

1 byte copied, 0.0008733 s, 1.1 kB/s 1+0 records in
1+0 records out

1 byte copied, 0.0008764 s, 1.1 kB/s
1+0 records in
1+0 records out

1 byte copied, 0.0008728 s, 1.1 kB/s 1+0 records in
1+0 records out

1 byte copied, 0.0008812 s, 1.1 kB/s 1+0 records in
1+0 records out

1 byte copied, 0.0008815 s, 1.1 kB/s 1+0 records in
1+0 records out

1 byte copied, 0.0007812 s, 1.3 kB/s 1+0 records in
1+0 records out

1 byte copied, 0.00094 s, 1.1 kB/s 1+0 records in
1+0 records out

1 byte copied, 0.0014346 s, 0.7 kB/s 1+0 records in
1+0 records out

1 byte copied, 0.0011726 s, 0.9 kB/s 1+0 records in
1+0 records out

1 byte copied, 0.0009611 s, 1.0 kB/s 1+0 records in
1+0 records out

1 byte copied, 0.000976 s, 1.0 kB/s 1+0 records in
1+0 records out

1 byte copied, 0.000998 s, 1.0 kB/s 1+0 records in
1+0 records out

1 byte copied, 0.0009959 s, 1.0 kB/s 1+0 records in
1+0 records out

1 byte copied, 0.0010128 s, 1.0 kB/s 1+0 records in
1+0 records out

1 byte copied, 0.0008609 s, 1.2 kB/s 1+0 records in
1+0 records out

1 byte copied, 0.0010422 s, 1.0 kB/s 1+0 records in
1+0 records out

1 byte copied, 0.0010409 s, 1.0 kB/s 1+0 records in

1+0 records out
1 byte copied, 0.0007966 s, 1.3 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0011194 s, 0.9 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.000884 s, 1.1 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0011693 s, 0.9 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0008168 s, 1.2 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0009581 s, 1.0 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0009555 s, 1.0 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0008653 s, 1.2 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.001148 s, 0.9 kB/s
1+0 records in
1+0 records out
1 byte copied, 0.0007973 s, 1.3 kB/s
1+0 records in
1+0 records out
1 byte copied, 0.0012466 s, 0.8 kB/s
1+0 records in
1+0 records out
1 byte copied, 0.0008711 s, 1.1 kB/s
1+0 records in
1+0 records out
1 byte copied, 0.00079 s, 1.3 kB/s
1+0 records in
1+0 records out
1 byte copied, 0.0007778 s, 1.3 kB/s
1+0 records in
1+0 records out
1 byte copied, 0.0008608 s, 1.2 kB/s
1+0 records in
1+0 records out
1 byte copied, 0.0009707 s, 1.0 kB/s
1+0 records in
1+0 records out
1 byte copied, 0.0010964 s, 0.9 kB/s
1+0 records in
1+0 records out
1 byte copied, 0.0008924 s, 1.1 kB/s
Сүффикс = XclL
ucvim@UCvim-PC:/mnt/c/Users/UCvim/Google Drive/MAI/FI & VS/Nikulin/labs/l21/test\$ ls
10AtGnST.OdZ 6f1o.XclL JA9Pn12o.WJn Q91UmkWn.ygP Wxrg0WMQ.J0p cywz.XclL gRic.XclL
tpXu1caz.SRN
mQRd.8p1
19v8.XclL 6t5I.MOY KQDh.XclL QoS1.2RK X3OE.XclL enEx.XclL gRic.eUx mQRd.XclL
1Uwo.XclL 6t5I.XclL KufkyGUM.2X7 QoS1.XclL ZstV.XclL enEx.IB1 hkKl.Hx1 mXlb.XclL uPspKhFB.n65

29UD.XclL 73VE.XclL L1Dk.XclL RZzu.XclL b6W8Lez7.5Eo fIxR.AMs hkKl.XclL myAn.XclL
wwMV.XclL
uEh5.XclL
2lkO.XclL 9qZK5ZM4.SMC MPBSbdS8.JL1 S1SS.XclL xICW.XclL
3Ej65h2f.xAy 9uVW1CA3.IBX Mx82ACKI.leu SDKi.XclL y4CGFeUJ.XgB
4PjyhUXM.bIw AOBxjJyu.9OM N3y4zP95.Tku TdvT.XclL yhxo7uyJ.apJ
5VeHp7th.XJq FVHAwRIY.2vz Omfkqc5a.RuW VtSX.XclL zqo61ytX.yJL
bE7K.XclL fIxR.XclL i64X7CYX.aUR ns8Y.XclL
bcNDWp5A.h8b fLBr.XclL jHFgvwNE.LDT ns8Y.zDc
c5fU.XclL fLBr.hij l4nzQkN2.vgT osNqxvsv.TcQ
cR6FZZIn.C2y fW9r.4L9 lKAFLVlx.qZY p3tOinVN.rty
5hR3.XclL Fu4y.XclL Q75qOkpf.uvt WoQbe9Uj.vC6 cywz.UUi fW9r.XclL lTeMma4G.kKN tCfdVogg.RvG
zsfx.XclL
ucvim@UCvim-PC:/mnt/c/Users/UCvim/Google Drive/MAI/FI & VS/Nikulin/labs/l21/test\$./l21.sh ?
Программа получает на вход суффикс, далее она создаёт список файлов temp с указанным суффиксом.
Список temp обрабатывается так, что в нём остаются только имена файлов. Далее создаётся список other из файлов
с отличными суффиксами от указанного. После чего программа заходит в цикл по списку temp и при
наличии хотя бы
одного совпадающего имени у temp и other удаляет все файлы с исходным суффиксом.
ucvim@UCvim-PC:/mnt/c/Users/UCvim/Google Drive/MAI/FI & VS/Nikulin/labs/l21/test\$./l21.sh -s XclL
Создание списка файлов temp с указанным суффиксом .XclL:
.19v8.XclL
.1Uwo.XclL ./29UD.XclL ./2lkO.XclL ./5hR3.XclL ./6f1o.XclL ./6t5I.XclL ./73VE.XclL ./bE7K.XclL ./
c5fU.XclL ./cywz.XclL ./enEx.XclL ./fIxR.XclL ./Fu4y.XclL ./fW9r.XclL ./gRic.XclL ./
hkKl.XclL ./KQDh.XclL
.L1Dk.XclL
.mQRd.XclL
.mXlb.XclL
.myAn.XclL
.ns8Y.XclL
.QoSsi.XclL
.RZzu.XclL
.S1SS.XclL
.SDKi.XclL
.TdvT.XclL
.uEh5.XclL
.VtSX.XclL
.wwMV.XclL
.X3OE.XclL
.xICW.XclL
.zsfx.XclL
.ZstV.XclL
Удаление ./ в начале и .XclL в конце для temp (необходимо для сравнения имён): 19v8
1Uwo 29UD 2lkO 5hR3 6f1o 6t5I 73VE bE7K c5fU cywz enEx fIxR fLBr Fu4y fW9r gRic
hkKl
KQDh
L1Dk
mQRd
mXlb
myAn
ns8Y

QoS
RZzu
S1SS
SDKi
TdvT
uEh5
VtSX
wwMV
X3OE
xICW
zsfx
ZstV

Создание списка файлов other с суффиксами, отличными от .XclL ./10AtGnST.OdZ
.3Ej65h2f.xAy ./4PjyhUXM.bIw ./5VeHp7th.XJq ./6t5I.MOY ./9qZK5ZM4.SMC ./9uVW1CA3.IBX ./
AOBxJyu.9OM ./b6W8Lez7.5Eo ./bcNDWp5A.h8b ./cR6FZZIn.C2y ./cywz.UUi ./enEx.lB1 ./flxR.AMs ./
fLBr.hij
.FVH AwRIY.2vz
.fW9r.4L9
.gRic.eUx
.hkKl.Hx1
.i64X7CYX.aUR
.JA9Pn12o.WJn
.jHFgvwNE.LDT
.KufkyGUM.2X7
.l4nzQkN2.vgT
.KAFLVlx.qZY
.lTeMma4G.kKN
.MPBSbdS8.JL1
.mQRd.8p1
.Mx82ACKI.leu
.N3y4zP95.Tku
.ns8Y.zDc
.Omfkqc5a.RuW
.osNqxvsv.TcQ
.p3tOinVN.rty
.Q75qOkpf.uvt
.Q91UmkWn.ygP
.QoS i.2RK
.tCfdVogg.RvG
.tpXu1caz.SRN
.uPspKhFB.n65
.WoQbe9Uj.vC6
.Wxrg0WMQ.J0p
.y4CGFeUJ.XgB
.yhxo7uyJ.apJ
.zqo61ytX.yJL

Удаление всех файлов с .XclL при наличии хотя бы одного совпадающего имени Чистка мусора:
удаление списков temp и other ucvim@UCvim-PC:/mnt/c/Users/UCvim/Google Drive/MAI/FI & VS/
Nikulin/labs/l21/test\$ ls

10AtGnST.OdZ 9qZK5ZM4.SMC KufkyGUM.2X7 Q75qOkpf.uvt b6W8Lez7.5Eo flxR.AMs
i64X7CYX.aUR mQRd.8p1 tpXu1caz.SRN

3Ej65h2f.xAy 9uVW1CA3.IBX MPBSbdS8.JL1 Q91UmkWn.ygP bcNDWp5A.h8b fLBr.hij jHFgvwNE.LDT
ns8Y.zDc uPspKhFB.n65
4PjyhUXM.bIw AOBxjJyu.9OM Mx82ACKI.leu QoS.i.2RK cR6FZZIn.C2y fW9r.4L9 l4nzQkN2.vgT
osNqvxsv.TcQ y4CGFeUJ.XgB
5VeHp7th.XJq FVHAwRIY.2vz N3y4zP95.Tku WoQbe9Uj.vC6 cywz.UUi gRic.eUx lKAFLVlx.qZY
p3tOinVN.rty yhxo7uyJ.apJ
6t5I.MOY JA9Pn12o.WJn Omfkqc5a.RuW Wxrg0WMQ.J0p enEx.lB1 hkK1.Hx1 lTeMma4G.kKN
tCfdVogg.RvG zqo61ytX.yJL
ucvim@UCvim-PC:/mnt/c/Users/UCvim/Google Drive/MAI/FI & VS/Nikulin/labs/l21/test rm *
ucvim@UCvim-PC:/mnt/c/Users/UCvim/Google Drive/MAI/FI & VS/Nikulin/labs/l21/test\$ ls
ucvim@UCvim-PC:/mnt/c/Users/UCvim/Google Drive/MAI/FI & VS/Nikulin/labs/l21/test\$./tester.sh 1+0
records in
1+0 records out
1 byte copied, 0.0009517 s, 1.1 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0010278 s, 1.0 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0008725 s, 1.1 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0009665 s, 1.0 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0009529 s, 1.0 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0010893 s, 0.9 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0012784 s, 0.8 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0011792 s, 0.8 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0009995 s, 1.0 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0010526 s, 1.0 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0009265 s, 1.1 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0012387 s, 0.8 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0009708 s, 1.0 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0009501 s, 1.1 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0009361 s, 1.1 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0010928 s, 0.9 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0009646 s, 1.0 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0009667 s, 1.0 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0009883 s, 1.0 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0010888 s, 0.9 kB/s 1+0 records in

1+0 records out
1 byte copied, 0.001053 s, 0.9 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0010372 s, 1.0 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0009279 s, 1.1 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0009558 s, 1.0 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.000961 s, 1.0 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0009076 s, 1.1 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0010497 s, 1.0 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0013002 s, 0.8 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0009238 s, 1.1 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0009564 s, 1.0 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0009501 s, 1.1 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0009538 s, 1.0 kB/s
1+0 records in
1+0 records out
1 byte copied, 0.0011016 s, 0.9 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0009684 s, 1.0 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0009358 s, 1.1 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0009098 s, 1.1 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0010518 s, 1.0 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0009914 s, 1.0 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0013836 s, 0.7 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.001064 s, 0.9 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0013266 s, 0.8 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0009792 s, 1.0 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0008237 s, 1.2 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0008858 s, 1.1 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0008884 s, 1.1 kB/s 1+0 records in

1+0 records out
1 byte copied, 0.0010583 s, 0.9 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.000858 s, 1.2 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0008082 s, 1.2 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0008249 s, 1.2 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0010064 s, 1.0 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.000774 s, 1.3 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0008173 s, 1.2 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0009455 s, 1.1 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0008556 s, 1.2 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0008419 s, 1.2 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0008001 s, 1.2 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0007488 s, 1.3 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0007827 s, 1.3 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0007822 s, 1.3 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0008552 s, 1.2 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0006924 s, 1.4 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0008235 s, 1.2 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0008146 s, 1.2 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.000737 s, 1.4 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.000836 s, 1.2 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0007593 s, 1.3 kB/s 1+0 records in
1+0 records out
1 byte copied, 0.0008136 s, 1.2 kB/s
Суффикс = FhrTdX
ucvim@UCvim-PC:/mnt/c/Users/UCvim/Google Drive/MAI/FI & VS/Nikulin/labs/l21/test\$ ls
0p.FhrTdX CK.CHr G9.FhrTdX OQ4OU5In.odt UJgRiHJd.hf9 ZeYjHc3r.sDc geaaEaG7.XGr
mQCKRQbu.l9C rP.FhrTdX x1.FhrTdX
5N.FhrTdX CK.FhrTdX I2dPKXw2.eYC OxehsXEZ.CLQ VY.FhrTdX a2.FhrTdX haLvl1dj.UVd
mY.FhrTdX th.FhrTdX xg.FhrTdX
5d.FhrTdX CTSKGPN9.dUo J7.FhrTdX v9.FhrTdX xg.QBQ
QG.EOn W1BLw0Z8.S4F c4uCbNV5.tmx hz.FhrTdX QG.FhrTdX X5.FhrTdX eQpNCUdx.1Z3 jm.FhrTdX

R0LmMilc.Taq Xd.FhrTdX eZ.FhrTdX lh.FhrTdX

ms.FhrTdX

ni.FhrTdX

o8lrWydt.r3i

6c.FhrTdX EG.FhrTdX vNMCCklh.zEt

A6QlAw5V.M7s En.FhrTdX vZ.FhrTdX

BH9WCOEE.IWD En.jl9 pkC4rKSg.7pp vZ.zUB

J7.Jdh Lk.FhrTdX

OP.FhrTdX Tg02QsF3.9ym YWQ7WGF0.rmK g1.FhrTdX m873qUkp.akr

Bt.FhrTdX G9.6Y4 OP.S84 Tj.FhrTdX Yo.FhrTdX g3.FhrTdX mO8F98LQ.5S6 qdSsuwiQ.wBi

wkDycP9E.pmh

ucvim@UCvim-PC:/mnt/c/Users/UCvim/Google Drive/MAI/FI & VS/Nikulin/labs/l21/test\$./l21.sh -s

FhrTdX Создание списка файлов temp с указанным суффиксом .FhrTdX:

./0p.FhrTdX

./5d.FhrTdX

./5N.FhrTdX

./6c.FhrTdX

./a2.FhrTdX

./Bt.FhrTdX

./CK.FhrTdX

./EG.FhrTdX

./En.FhrTdX

./eZ.FhrTdX

./g1.FhrTdX

./g3.FhrTdX

./G9.FhrTdX

./hz.FhrTdX

./J7.FhrTdX

./jm.FhrTdX

./lh.FhrTdX

./Lk.FhrTdX

./ms.FhrTdX

./mY.FhrTdX

./ni.FhrTdX

./OP.FhrTdX

./QG.FhrTdX

./rP.FhrTdX

./th.FhrTdX

./Tj.FhrTdX

./v9.FhrTdX

./VY.FhrTdX

./vZ.FhrTdX

./x1.FhrTdX

./X5.FhrTdX

./Xd.FhrTdX

./xg.FhrTdX

./Yo.FhrTdX

Удаление ./ в начале и .FhrTdX в конце для temp (необходимо для сравнения имён): 0p

5d 5N 6c a2 Bt CK EG En eZ g1 g3 G9 hz J7 jm lh Lk ms mY

ni

OP

QG

rP
th
Tj
v9
VY
vZ
x1
X5
Xd
xg
Yo

Создание списка файлов other с суффиксами, отличными от .FhrTdX ./A6QlAw5V.M7s
.BH9WCOEE.IWD ./c4uCbNV5.tmx ./CK.CHr ./CTSKGPN9.dUo ./En.jl9 ./eQpNCUdx.1Z3 ./G9.6Y4 ./
geaaEaG7.XGr ./haLvl1dj.UVd ./I2dPKXw2.eYC ./J7.Jdh ./m873qUkp.akr ./mO8F98LQ.5S6 ./
mQCKRQbu.l9C ./o8lrWydt.r3i ./OP.S84 ./OQ4OU5In.odt ./OxehsXEZ.CLQ ./pkC4rKSg.7pp
.qdSsuwiQ.wBi
.QG.EOn
.R0LmMilc.Taq
.Tg02QsF3.9ym
.UJgRiHJd.hf9
.vNMCCklh.zEt
.vZ.zUB
.W1BLw0Z8.S4F
.wkDycP9E.pmh
.xg.QBQ
.YWQ7WGF0.rmK
.ZeYjHc3r.sDc

Удаление всех файлов с .FhrTdX при наличии хотя бы одного совпадающего имени

Чистка мусора: удаление списков temp и other

ucvim@UCvim-PC:/mnt/c/Users/UCvim/Google Drive/MAI/FI & VS/Nikulin/labs/l21/test\$ ls
A6QlAw5V.M7s En.jl9 OP.S84 R0LmMilc.Taq YWQ7WGF0.rmK geaaEaG7.XGr mQCKRQbu.l9C
vNMCCklh.zEt

BH9WCOEE.IWD G9.6Y4 OQ4OU5In.odt Tg02QsF3.9ym ZeYjHc3r.sDc haLvl1dj.UVd o8lrWydt.r3i
vZ.zUB

CK.CHr I2dPKXw2.eYC OxehsXEZ.CLQ UJgRiHJd.hf9 c4uCbNV5.tmx m873qUkp.akr pkC4rKSg.7pp
wkDycP9E.pmh

CTSKGPN9.dUo J7.Jdh QG.EOn W1BLw0Z8.S4F eQpNCUdx.1Z3 mO8F98LQ.5S6 qdSsuwiQ.wBi
xg.QBQ

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.	Дат а	Врем я	Событие	Действие по исправлению	Примечание

--	--	--	--	--	--

10.Замечания автора по существу работы: -

11.Выводы: я составил программу выполнения заданных действий над файлами на одном из интерпретируемых командных языков ОС UNIX

Подпись студента
