

# Отчет по лабораторной работе №21

по курсу: 1 фундаментальная информатика

студента группы М80-101Б-21 Тулина Ивана, № по списку: 22

Контакты www, e-mail, icq, skype: i.tulin0107@gmail.com

Работа выполнена: «30» апреля 2022г.

Преподаватель: Титов В. К. каф. 806

Входной контроль знаний с оценкой \_\_\_\_\_

Отчет сдан «    » \_\_\_\_\_ 201 \_\_\_\_ г., итоговая оценка \_\_\_\_\_

Подпись преподавателя \_\_\_\_\_

1 **Тема:** Программирование на интерпретируемых языках.

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

3 **Задание (вариант № 22):** Замена для всех файлов, имеющих размер меньше заданного, суффиксов имен на первые литеры имен файлов.

4 **Оборудование (лабораторное):**  
ЭВМ \_\_\_\_\_ - \_\_\_\_\_, процессор \_\_\_\_\_ - \_\_\_\_\_, имя узла сети \_\_\_\_\_ - \_\_\_\_\_ с ОП \_\_\_\_\_ - \_\_\_\_\_ Мб, НМД \_\_\_\_\_ Мб. Терминал \_\_\_\_\_ адрес \_\_\_\_\_. Принтер \_\_\_\_\_  
Другие устройства \_\_\_\_\_

*Оборудование ПЭВМ студента, если использовалось:*

Процессор Intel Core i5-7300HQ с ОП 7,87 Мб, НМД 15360 Мб. Монитор: встроенный  
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 Ubuntu версия 20.04.3 LTS  
интерпретатор команд bash версия \_\_\_\_\_  
Система программирования \_\_\_\_\_ версия \_\_\_\_\_  
Редактор текстов Emacs версия 3.22.30  
Утилиты операционной системы \_\_\_\_\_  
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)

От программы требуется изменить суффиксы имен для всех находящихся в текущей директории файлов, чей объём меньше числа-границы, заданного при запуске сценария.

Алгоритм будет состоять в

- 1) считывании в качестве аргумента границы обработки;
- 2) измерении объёма, выделенного для хранения каждого из файлов в текущей директории;
- 3) сравнении полученных во 2-ом пункте значений объёма для каждого файла с установленной в 1-ом пункте границей;
- 4) замене суффикса имени файла первой буквой имени при условии, что объём проверяемого файла меньше установленной границы.

Для проверки корректности работы программы можно использовать утилиты ls и du.

- Число-граница по умолчанию должна задаваться в байтах, но при указании параметров: b (байты), K (килобайты), M (мегабайты) или G (гигабайты) – граница обработки должна считываться в соответствующих единицах измерения.
- При отсутствии обязательного параметра – числа-границы – программа должна запросить его.
- При запуске сценария с неправильно введенными параметрами программа должна вывести сообщение об ошибке в синтаксисе, с указанием поля, в котором допущена ошибка.
- При запуске скрипта с параметром «?» программа должна вывести подсказку (свою спецификацию)

Итого, большую часть текста сценария составит ветвление учитывающее возможные варианты ввода параметров

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

Текст программы myscript:

```
#!/bin/bash

if [ -z $1 ]; then
echo 'myscript: пропущен обязательный операнд: число, задающее границу обработки'
echo 'Запустите "~/myscript ?" для получения более подробного описания'
exit

elif [ $1 = "?" ] && [ -z "$2" ]; then
echo -e "\nСпецификация программы:\n\t Замена для всех файлов, имеющих размер меньше заданного,\n\t суффиксов имен на первые литеры имен файлов."
echo -e "\nДопустимый синтаксис: ~/myscript [ПАРАМЕТР] [ЧИСЛО]"
echo -e " ЧИСЛО (обязательный аргумент) - верхняя граница размера файлов,\n\t для которых будет произведена замена суффикса"
echo -e " ПАРАМЕТР (необязательный аргумент) Задаёт единицы измерения,\n\t в которых устанавливается граница (по умолчанию используются байты)\n\t b - байты\n\t K - Кб\n\t M - Мб\n\t G - Гб"
exit

elif [ "$1" = b ] || [ "$1" = K ] || [ "$1" = M ] || [ "$1" = G ]; then
if [ -n "$2" ]; then
echo $2 | tr -d '0-9' ]; then
echo -e "myscript: недопустимый синтаксис: ~/myscript $1 $2\n\t ^..границу задает целое число"
exit
elif [ "$1" = b ]; then lim=$2
elif [ "$1" = K ]; then lim=`expr $2 \* 1024`
elif [ "$1" = M ]; then lim=`expr $2 \* 1024 \* 1024`
elif [ "$1" = G ]; then lim=`expr $2 \* 1024 \* 1024 \* 1024`
fi

elif [ -z "$2" ]; then lim=$1
else
if [ -z "$3" ]; then
echo -e "myscript: неверно указан необязательный операнд: ~/myscript $1 $2\n\t несуществующий параметр..^"
echo 'Запустите "~/myscript ?" для получения более подробного описания'
else
echo -e "myscript: недоступный синтаксис: ~/myscript $1\n\t ^..границу задает целое число"
echo 'Запустите "~/myscript ?" для получения более подробного описания'
fi
exit
fi

for name in `ls`; do
if [ `du -b $name | awk '{print $1}' -le $lim` ]; then
suff=`echo $name | awk '{print substr($1, 1, 1)}`
temp=$name
while [ -n "`echo $temp | grep '\.'`" ]; do
temp=`echo $temp | sed s/\./`
done
if [ $name != `echo $temp | sed s/$/\`.$suff`/^ ]; then
mv $name `echo $temp | sed s/$/\`.$suff`/^
fi
fi
done
```

Проведение тестов:

- 1) cd ~/tempdir/ # перейти в директорию, заготовленную для проведения тестов сценария (директория должна содержать файлы и каталоги с разным объемом и разными расширениями)
- 2) ls # вывести список файлов текущей директории
- 3) du -b /\* # выяснить объём этих файлов
- 5) ~/myscript # запуск скрипта без параметров (программа должна запросить обязательный параметр)
- 6) ~/myscript ? # запуск скрипта с параметром «?» (программа выведет подсказку со спецификацией, допустимым синтаксисом и описанием параметров ввода)
- 7) ~/myscript 94qwe34rt77y uior1234 67z # запуск скрипта с некорректно введенными параметрами  
# (программа должна выдать комментированное сообщение об ошибке и указать на первое поле)

- 8) ~/myscript b gh75jk8      # запуск скрипта с некорректно введенным вторым параметром (программа  
# должна выдать комментированное сообщение об ошибке и указать на второе поле)
- 9) ~/myscript Luytbj 1024    # запуск скрипта с некорректно введенным первым параметром (программа должна  
# выдать комментированное сообщение об ошибке и указать на первое поле)
- 10) ~/myscript 102400        # запуск скрипта без необязательных параметров
- 11) ls                         # проверка произведенных изменений
- 12) ~/myscript K 1024        # запуск скрипта с границей, заданной в килобайтах
- 13) ls

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

*Допущен к выполнению работы. Подпись преподавателя \_\_\_\_\_*

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

```
yusayu@YS:~$ cat myscript
#!/bin/bash
if [ -z $1 ]; then
    echo 'myscript: пропущен обязательный операнд: число, задающее границу обработки'
    echo 'Запустите "~/myscript ?" для получения более подробного описания'
    exit
elif [ $1 = "?" ] && [ -z "$2" ]; then
    echo -e "\nСпецификация программы:\n\t Замена для всех файлов, имеющих размер меньше заданного,\n\t суффиксов имен на первые литеры имен файлов."
    echo -e "\nДопустимый синтаксис: ~/myscript [ПАРАМЕТР] [ЧИСЛО]"
    echo -e " ЧИСЛО (обязательный аргумент) - верхняя граница размера файлов,\n\t для которых будет произведена замена суффикса"
    echo -e " ПАРАМЕТР (необязательный аргумент) Задаёт единицы измерения,\n\t в которых устанавливается граница (по умолчанию используются байты)\n\t b - байты\n\t K - Кб\n\t M - Мб\n\t G - Гб"
    exit
elif [ "$1" = b ] || [ "$1" = K ] || [ "$1" = M ] || [ "$1" = G ]; then
    if [ -n "`echo $2|tr -d '0-9'`" ]; then
        echo -e "myscript: недопустимый синтаксис: ~/myscript $1 $2\n\t ^..границу\n\t задает целое число"
        exit
    elif [ "$1" = b ]; then lim=$2
    elif [ "$1" = K ]; then lim=`expr $2 \* 1024`
    elif [ "$1" = M ]; then lim=`expr $2 \* 1024 \* 1024`
    elif [ "$1" = G ]; then lim=`expr $2 \* 1024 \* 1024 \* 1024`
    fi
    elif [ -z "`echo $1|tr -d '0-9'`" ]; then lim=$1
    else
        if [ -z "`echo $2|tr -d '0-9'`" ]; then
            echo -e "myscript: неверно указан необязательный операнд: ~/myscript $1 $2\n\t несуществующий параметр..^"
            echo 'Запустите "~/myscript ?" для получения более подробного описания'
        else
            echo -e "myscript: недоступный синтаксис: ~/myscript $1\n\t ^..границу задает\n\t целое число"
            echo 'Запустите "~/myscript ?" для получения более подробного описания'
        fi
    fi
    exit
fi

for name in `ls`; do
    if [ `du -b $name|awk '{print $1}' -le $lim` ]; then
        suff=`echo $name |awk '{print substr($1, 1, 1)}`
        temp=$name
        while [ -n "`echo $temp|grep '\.'`" ]; do
            temp=`echo $temp |sed s/\./^`
        done
        if [ $name != `echo $temp |sed s/s/"\.$suff"/` ]; then
            mv $name `echo $temp |sed s/s/"\.$suff"/`
        fi
    fi
done

yusayu@YS:~$ cd ~/tempdir/
yusayu@YS:~/tempdir$ ls
lab8 pic.pdf prac2007.djvu лекция_ЭВМ
yusayu@YS:~/tempdir$ du -b .*
4499 ./lab8
463752 ./pic.pdf
175883 ./prac2007.djvu
```

6999 ./лекция\_ЭВМ

yusayu@YS:~/tempdir\$ ~/myscript

myscript: пропущен обязательный операнд: число, задающее границу обработки

Запустите "~/myscript ?" для получения более подробного описания

yusayu@YS:~/tempdir\$ ~/myscript ?

Спецификация программы:

Замена для всех файлов, имеющих размер меньше заданного,  
суффиксов имен на первые литеры имен файлов.

Допустимый синтаксис: ~/myscript [ПАРАМЕТР] [ЧИСЛО]

ЧИСЛО (обязательный аргумент) - верхняя граница размера файлов,  
для которых будет произведена замена суффикса

ПАРАМЕТР (необязательный аргумент) Задаёт единицы измерения,  
в которых устанавливается граница (по умолчанию используются байты)

b - байты

K - Kб

M - Mб

G - Gб

yusayu@YS:~/tempdir\$ ~/myscript 94qwe34rt77y uiop1234 67z

myscript: недоступный синтаксис: ~/myscript 94qwe34rt77y

^..границу задает целое число

Запустите "~/myscript ?" для получения более подробного описания

yusayu@YS:~/tempdir\$ ~/myscript b gh75jk8

myscript: недопустимый синтаксис: ~/myscript b gh75jk8

^..границу задает целое число

yusayu@YS:~/tempdir\$ ~/myscript Luytbj 1024

myscript: неверно указан необязательный операнд: ~/myscript Luytbj 1024

несуществующий параметр..^

Запустите "~/myscript ?" для получения более подробного описания

yusayu@YS:~/tempdir\$ ~/myscript 102400

yusayu@YS:~/tempdir\$ ls

lab8.l pic.pdf prac2007.djvu лекция\_ЭВМ.л

yusayu@YS:~/tempdir\$ ~/myscript K 1024

yusayu@YS:~/tempdir\$ ls

lab8.l pic.p prac2007.p лекция\_ЭВМ.л

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

10 **Замечания автора** по существу работы \_\_\_\_\_

---



---



---

#### 11 Выводы

В процессе работы я получил навыки составления сценариев обработки файлов в командной оболочке bash

---



---



---



---



---



---



---



---



---



---

Недочёты при выполнении задания могут быть устранены следующим образом: \_\_\_\_\_

---



---



---



---

Подпись студента \_\_\_\_\_