|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ «ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ»  
КАФЕДРА «ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ЭВМ И ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ»

**ОТЧЕТ ПО УЧЕБНОЙ ПРАКТИКЕ**

Студент **- Нгуен Ань Зунг**

Группа **ИУ7И-11Б**

Тип практики **Проектно-технологическая практика.**

Название предприятия **НУК ИУ МГТУ им. Н. Э. Баумана**

Студент **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ - Нгуен Ань Зунг.**

Руководитель практики \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ **Ломовской И. В.**

Руководитель практики \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ **Кострицкий А. С.**

Оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

*2022 г.*

ОГЛАВЛЕНИЕ

ВВЕДЕНИИ ............................................................................................................3

РАЗЛИЧНЫЕ ТЕСТОВЫЕ СЛУЧАИ ..................................................................5

ЗАКЛЮЧЕНИЕ.......................................................................................................6

ЛИСТИНГ ИСХОДНОГО КОДА ..........................................................................8

ВВЕДЕНИИ

Разработать скрипт командной оболочки для сравнения содержимого двух тек- стовых файлов по определённым правилам.

Варианты сравнения

1. Сравниваются последовательности целых чисел в файлах. Каждое целое число заведомо входит в диапазон знакового целого в 4 байта. Целые чис- ла отделяются от других символов в файле пробельными символами. Об- ращаем Ваше внимание, что в файле могут находиться не только числа, но вычленять и анализировать нужно именно их. Сравниваются числа как текст — числа 100 и 0100 следует считать различными.

Название скрипта: comparator1.sh.

1. Сравнивается текст в файлах после первого вхождения подстроки «string:». Подразумевается строгое сравнение с учётом разницы в про- бельных символах и символах окончания строки.

Название скрипта: comparator2.sh.

1. Сравниваются последовательности чисел с плавающей точкой (ЧПТ), за- писанных **не** в экспоненциальной форме. Каждое найденное ЧПТ заведо- мо входит в диапазон чисел двойной точности. Обращаем Ваше внима- ние, что в файле могут находиться не только числа, но вычленять и ана- лизировать нужно именно их. Сравниваются числа как текст — напри- мер, числа 1.0 и 1.00 следует считать различными.

Название скрипта: comparator3.sh.

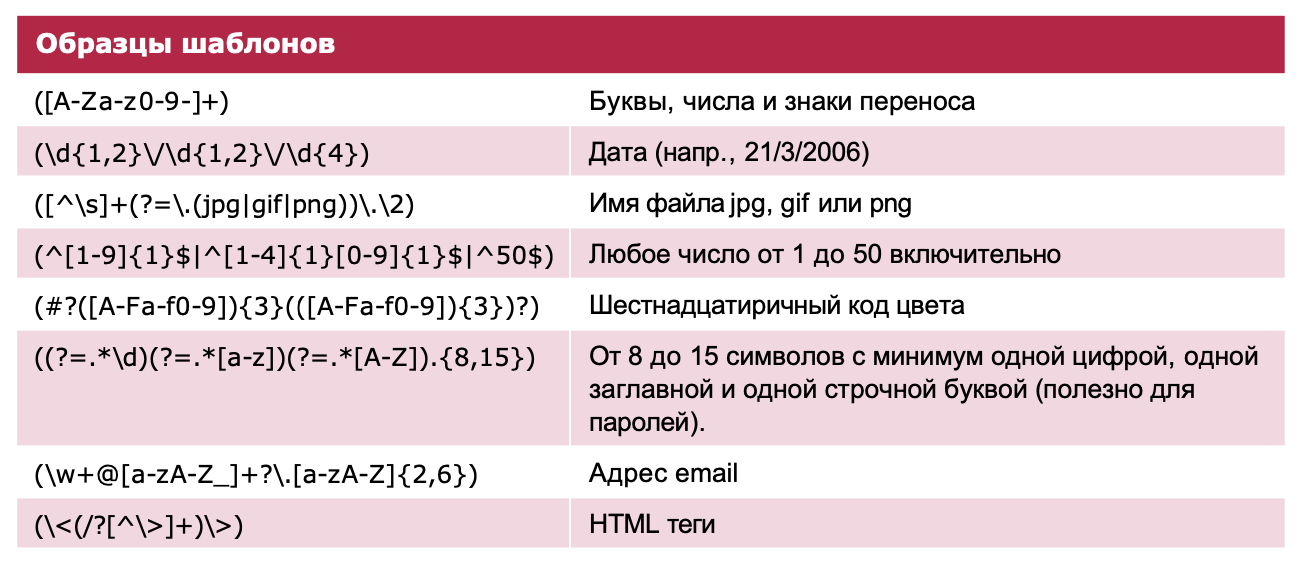
1. Сравниваются последовательности чисел с плавающей точкой (ЧПТ), за-

писанных **в том числе** в экспоненциальной форме. Каждое найденное ЧПТ заведомо входит в диапазон чисел двойной точности. Обращаем Ваше внимание, что в файле могут находиться не только числа, но вычле- нять и анализировать нужно именно их. Сравниваются числа как текст — например, числа 1.0 и 1.00 следует считать различными, числа 1.0 и 1.0e0 – тоже.

Название скрипта: comparator4.sh.

РАЗЛИЧНЫЕ ТЕСТОВЫЕ СЛУЧАИ

1. Некоторые примеры «регулярных регулярные выражение», использованные в статье:



1. Как пользоваться "Право доступа" использованным в статье

* **r** — read (чтение) — право просматривать содержимое файла;
* **w** — write (запись) — право изменять содержимое файла;
* **x** — execute (выполнение) — право запускать файл, если это программа или скрипт.

1. Некоторые способы использования cmp в Linux
   1. Сравнение содержимого двух файлов

* $ cmp [имя-файла-1] [имя-файла-2]
  1. Вывод отличающихся значений байтов
* $ cmp -b [имя-файла-1] [имя-файла-2]
  1. Пропуск начальных байтов двух файлов
* $ cmp -i [количество-пропускаемых-байтов] [имя-файла-1] [имя-файла-2]
  1. Вывод информации о позициях (и значениях) всех отличающихся байтов
* $ cmp -l [имя-файла-1] [имя-файла-2]
  1. Сокрытие вывода
* $ cmp -s [имя-файла-1] [имя-файла-2]
  1. Ограничение количества байтов для сравнения
* $ cmp -n [количество-байтов-для-сравнения] [имя-файла-1] [имя-файла-2]

ЗАКЛЮЧЕНИЕ

1. Comparator1.sh

Чтобы сравнить целые числа в двух файлах, мы сначала проверим, существует ли файл или нет. Если файл не существует, будет напечатан экран «Не удается найти файл». Если файл существует, мы продолжим проверять права доступа к файлу. Далее мы будем запускать каждую строку в файле и использовать «регулярные выражения», чтобы найти целое число. Затем мы будем использовать «cmp -s» для сравнения двух файлов. Когда 2 файла одинаковы, он будет печатать «файлы совпадают» и, наоборот, печатать «файлы не совпадают».

1. Comparator3.sh

Чтобы сравнить числа с плавающей точкой в диапазон чисел двойной точности в двух файлах, мы сначала проверим, существует ли файл или нет. Если файл не существует, будет напечатан экран «Не удается найти файл». Если файл существует, мы продолжим проверять права доступа к файлу. Далее мы будем запускать каждую строку в файле и использовать «регулярные выражения», чтобы найти целое число. Затем мы будем использовать «cmp -s» для сравнения двух файлов. Когда 2 файла одинаковы, он будет печатать «файлы совпадают» и, наоборот, печатать «файлы не совпадают».

1. Comparator4.sh

Чтобы сравнить последовательности чисел с плавающей точкой (ЧПТ), записанных в том числе в экспоненциальной форме в двух файлах, мы сначала проверим, существует ли файл или нет. Если файл не существует, будет напечатан экран «Не удается найти файл». Если файл существует, мы продолжим проверять права доступа к файлу. Далее мы будем запускать каждую строку в файле и использовать «регулярные выражения», чтобы найти целое число. Затем мы будем использовать «cmp -s» для сравнения двух файлов. Когда 2 файла одинаковы, он будет печатать «файлы совпадают» и, наоборот, печатать «файлы не совпадают».

ЛИСТИНГ ИСХОДНОГО КОДА

N1.

#!/bin/bash

if [ ! -f $1 ]; then

echo Ошибка! Файл 1 не найден!

exit 2

fi

if [ ! -f $2 ]; then

echo Ошибка! Файл 2 не найден!

exit 2

fi

if [ ! -r "$1" ]; then

echo Ошибка! У вас нет прав доступа к файлу 1!

exit 2

fi

if [ ! -r "$2" ]; then

echo Ошибка! У вас нет прав доступа к файлу 2!

exit 2

fi

file1=$(cat $1)

file2=$(cat $2)

file1\_nums=''

for word in $file1; do

echo $word

if [[ "$word" =~ ^[-+]?[0-9]+$ ]]; then

file1\_nums="$file1\_nums $word"

echo file1\_nums

fi

done

file2\_nums=''

for word in $file2; do

echo $word

if [[ "$word" =~ ^[-+]?[0-9]+$ ]]; then

file2\_nums="$file2\_nums $word"

echo file2\_nums

fi

done

if cmp -s "$file1\_nums" "$file2\_nums"; then

echo Файлы совпадают

exit 0

else

echo Файлы не совпадают

exit 1

fi

N3.

#!/bin/bash

if [ ! -f $1 ]; then

echo Ошибка! Файл 1 не найден!

exit 2

fi

if [ ! -f $2 ]; then

echo Ошибка! Файл 2 не найден!

exit 2

fi

if [ ! -r "$1" ]; then

echo Ошибка! У вас нет прав доступа к файлу 1!

exit 2

fi

if [ ! -r "$2" ]; then

echo Ошибка! У вас нет прав доступа к файлу 2!

exit 2

fi

file1=$(cat $1)

file2=$(cat $2)

file1\_nums=''

for word in $file1; do

echo $word

if [[ "$word" =~ ^[+-]?[0-9]\*[.,]?[0-9]{1,2}$ ]] ; then

file1\_nums="$file1\_nums $word"

echo file1\_nums

fi

done

file2\_nums=''

for word in $file2; do

echo $word

if [[ "$word" =~ ^[+-]?[0-9]\*[.,]?[0-9]{1,2}$ ]] ; then

file2\_nums="$file2\_nums "$word"

echo file2\_nums

fi

done

if cmp -s "$file1\_nums" "$file2\_nums"; then

echo Файлы совпадают

exit 0

else

echo Файлы не совпадают

exit 1

fi

N4.

#!/bin/bash

if [ ! -f $1 ]; then

echo Ошибка! Файл 1 не найден!

exit 2

fi

if [ ! -f $2 ]; then

echo Ошибка! Файл 2 не найден!

exit 2

fi

if [ ! -r "$1" ]; then

echo Ошибка! У вас нет прав доступа к файлу 1!

exit 2

fi

if [ ! -r "$2" ]; then

echo Ошибка! У вас нет прав доступа к файлу 2!

exit 2

fi

file1=$(cat $1)

file2=$(cat $2)

file1\_nums=''

for word in $file1; do

echo $word

if [[ "$word" =~ ^[+-]?[0-9]\*[.,]?[0-9]{1,2}([eE][+-]?[0-9]+)?$ ]]; then

file1\_nums="$file1\_nums $word"

echo file1\_nums

fi

done

file2\_nums=''

for word in $file2; do

echo $word

if [[ "$word" =~ ^[+-]?[0-9]\*[.,]?[0-9]{1,2}([eE][+-]?[0-9]+)?$ ]]; then

file2\_nums="$file2\_nums $word"

echo file2\_nums

fi

done

if cmp -s "$file1\_nums" "$file2\_nums"; then

echo Файлы совпадают

exit 0

else

echo Файлы не совпадают

exit 1

fi