**Содержание**

[Введение 3](#_Toc122458859)

[Сценарий командной оболочки 4](#_Toc122458860)

[Тестирование сценариев командной оболочки 15](#_Toc122458861)

[Заключение 19](#_Toc122458862)

[Список литературы 20](#_Toc122458863)

# Введение

***Цель работы*** – Разработать скрипт командной оболочки для сравнения содержимого двух текстовых файлов по определённым правилам.

В первом варианте ведется сравнение последовательностей целых чисел в файлах. Каждое число заведомо входит в диапазон знакового целого в 4 байта. Целые числа отделяются от других символов в файлах пробельными символами. В файлах также могут встречаться не только числа, но при сравнении учитываются только они. Сравнение чисел аналогично сравнению текста – 100 и 0100 считаются различными.

Во втором варианте сравнивается текст в файлах после первого вхождения подстроки «string:». (строгое сравнение с учётом разницы в пробельных символах и символах окончания строки)

В третьем варианте сравниваются последовательности чисел с плавающей точкой (ЧПТ), записанных не в экспоненциальной форме. Каждое найденное ЧПТ заведомо входит в диапазон чисел двойной точности. В файлах также могут встречаться не только числа, но при сравнении учитываются только они. Сравнение чисел аналогично сравнению текста – 1.0 и 1.00 считаются различными.

В четвертом варианте сравниваются последовательности чисел с плавающей точкой (ЧПТ), записанных в том числе в экспоненциальной форме. Каждое найденное ЧПТ заведомо входит в диапазон чисел двойной точности. В файлах также могут встречаться не только числа, но при сравнении учитываются только они. Сравнение чисел аналогично сравнению текста – 1.0 и 1.00 считаются различными, числа 1.0 и 1.0e0 – тоже.

# Сценарий командной оболочки

**comparator1.sh:**

*#!/bin/bash*

**if** ! [[ "$#" = "2" || "$#" = "3" ]]; **then** *#проверка количества аргументов*

exit 1

**fi**

**if** ! [[ -e "$1" && -e "$1" ]]; **then** *#проверка существования файлов*

exit 1

**fi**

**if** ! [[ -f $1 && -f $2 ]]; **then**

exit 1

**fi**

*#проверка, что файл существует и является обычным файлом*

**if** ! [[ -r "$1" || -r "$2" ]]; **then**

exit 1

**fi**

*#проверка, что файл доступен для чтения*

**if** [[ "$#" = "3" ]]; **then**

**if** ! [[ "$3" = "-v" ]]; **then**

exit 1

**fi**

str1=""

str2=""

IFS\_OLD="$IFS"

count="0"

myfile1="$(mktemp)"

DONE=false

**until** $DONE ;**do**

read -r y || DONE=true

**for** i in $y;

**do**

**if** [[ "$i" =~ ^[+-]?[0-9][0-9]\*$ ]]; **then**

echo "$i " >> "$myfile1"

count=**$((**$count + 1**))**

**fi**

**done**

**done** < "$1"

**if** [[ $count == "0" ]]; **then**

rm "$myfile1"

exit 1

**fi**

*#Проверка наличия чисел в первом файле*

count="0"

myfile2="$(mktemp)"

DONE=false

**until** $DONE ;**do**

read -r y || DONE=true

**for** i in $y;

**do**

**if** [[ "$i" =~ ^[+-]?[0-9][0-9]\*$ ]]; **then**

echo "$i " >> "$myfile2"

count=**$((**$count + 1**))**

**fi**

**done**

**done** < "$2"

IFS="$IFS\_OLD"

**if** [[ $count == "0" ]]; **then**

rm "$myfile1" "$myfile2"

exit 1

**fi**

*#Проверка наличия чисел во втором файле*

count="0"

**if** cmp -s "$myfile1" "$myfile2" ; **then**

rm "$myfile1" "$myfile2"

echo "Files are similar"

exit 0

**else**

rm "$myfile1" "$myfile2"

echo "Files are not similar"

exit 1

**fi**

**else**

str1=""

str2=""

IFS\_OLD="$IFS"

DONE=false

count="0"

myfile1="$(mktemp)"

**until** $DONE ;**do**

read -r y || DONE=true

**for** i in $y;

**do**

**if** [[ "$i" =~ ^[+-]?[0-9][0-9]\*$ ]]; **then**

echo "$i " >> "$myfile1"

count=**$((**$count + 1**))**

**fi**

**done**

**done** < "$1"

**if** [[ $count == "0" ]]; **then**

rm "$myfile1"

exit 1

**fi**

*#Проверка наличия чисел в первом файле*

count="0"

myfile2="$(mktemp)"

DONE=false

**until** $DONE ;**do**

read -r y || DONE=true

**for** i in $y;

**do**

**if** [[ "$i" =~ ^[+-]?[0-9][0-9]\*$ ]]; **then**

echo "$i " >> "$myfile2"

count=**$((**$count + 1**))**

**fi**

**done**

**done** < "$2"

IFS="$IFS\_OLD"

**if** [[ $count == "0" ]]; **then**

rm "$myfile1" "$myfile2"

exit 1

**fi**

*#Проверка наличия чисел во втором файле*

count="0"

**if** cmp -s "$myfile1" "$myfile2" ; **then**

rm "$myfile1" "$myfile2"

exit 0

**else**

rm "$myfile1" "$myfile2"

exit 1

**fi**

**fi**

**comparator2.sh:**

*#!/bin/bash*

**if** ! [[ "$#" = "2" || "$#" = "3" ]]; **then** *#проверка количества аргументов*

exit 1

**fi**

**if** ! [[ -e "$1" && -e "$1" ]]; **then** *#проверка существования файлов*

exit 1

**fi**

**if** ! [[ -f $1 && -f $2 ]]; **then**

exit 1

**fi**

*#проверка, что файл существует и является обычным файлом*

**if** ! [[ -r "$1" || -r "$2" ]]; **then**

exit 1

**fi**

*#проверка, что файл доступен для чтения*

**if** [[ "$#" = "3" ]]; **then**

**if** ! [[ "$3" = "-v" ]]; **then**

exit 1

**fi**

*#Проверка наличия ключа -v*

str1=""

count=0

count1=0

count2=0

str2=""

IFS\_OLD="$IFS"

IFS=''

count="0"

myfile1="$(mktemp)"

DONE=false

**until** $DONE ;**do**

read -r y || DONE=true

**if** [[ $count == 0 ]]; **then**

**if** [[ "$y" == \*"string:"\* ]]; **then**

count1=**$((**$count1+1**))**

count=**$((**$count+1**))**

echo "$(echo "$y" | grep -Eo "string:.\*")" \

>> "$myfile1"

**fi**

**else**

count1=**$((**$count1+1**))**

echo "$y" >> "$myfile1"

**fi**

**done** < "$1"

**if** [[ $count == 0 ]]; **then**

rm "$myfile1"

echo "The 'string:' is not found"

exit 1

**fi**

*#Проверка наличия подстроки в первом файле*

count=0

myfile2="$(mktemp)"

DONE=false

**until** $DONE ;**do**

read -r y || DONE=true

**if** [[ $count == 0 ]]; **then**

**if** [[ "$y" == \*"string:"\* ]]; **then**

count2=**$((**$count2+1**))**

count=**$((**$count+1**))**

echo "$(echo "$y" | grep -Eo "string:.\*")" \ >> "$myfile2"

**fi**

**else**

count2=**$((**$count2+1**))**

echo "$y" >> "$myfile2"

**fi**

**done** < "$2"

**if** [[ $count == 0 ]]; **then**

rm "$myfile1" "$myfile2"

echo "The 'string:' is not found"

exit 1

**fi**

*#Проверка наличия подстроки во втором файле*

**if** ! [[ "$count1" == "$count2" ]]; **then**

rm "$myfile1" "$myfile2"

echo "Files are different"

exit 1

**fi**

count=0

IFS="$IFS\_OLD"

**if** cmp -s "$myfile1" "$myfile2" ; **then**

rm "$myfile1" "$myfile2"

echo "Files are similar"

exit 0

**else**

rm "$myfile1" "$myfile2"

echo "Files are different"

exit 1

**fi**

**else**

str1=""

count=0

count1=0

count2=0

str2=""

IFS\_OLD="$IFS"

IFS=''

count="0"

myfile1="$(mktemp)"

DONE=false

**until** $DONE ;**do**

read -r y || DONE=true

**if** [[ $count == 0 ]]; **then**

**if** [[ "$y" == \*"string:"\* ]]; **then**

count1=**$((**$count1+1**))**

count=**$((**$count+1**))**

echo "$(echo "$y" | grep -Eo "string:.\*")" \

>> "$myfile1"

**fi**

**else**

count1=**$((**$count1+1**))**

echo "$y" >> "$myfile1"

**fi**

**done** < "$1"

**if** [[ $count == 0 ]]; **then**

rm "$myfile1"

exit 1

**fi**

*#Проверка наличия подстроки в первом файле*

count=0

myfile2="$(mktemp)"

DONE=false

**until** $DONE ;**do**

read -r y || DONE=true

**if** [[ $count == 0 ]]; **then**

**if** [[ "$y" == \*"string:"\* ]]; **then**

count2=**$((**$count2+1**))**

count=**$((**$count+1**))**

echo "$(echo "$y" | grep -Eo "string:.\*")" \ >> "$myfile2"

**fi**

**else**

count2=**$((**$count2+1**))**

echo "$y" >> "$myfile2"

**fi**

**done** < "$2"

**if** [[ $count == 0 ]]; **then**

rm "$myfile1" "$myfile2"

exit 1

**fi**

*#Проверка наличия подстроки во втором файле*

**if** ! [[ "$count1" == "$count2" ]]; **then**

rm "$myfile1" "$myfile2"

exit 1

**fi**

count=0

IFS="$IFS\_OLD"

**if** cmp -s "$myfile1" "$myfile2" ; **then**

rm "$myfile1" "$myfile2"

exit 0

**else**

rm "$myfile1" "$myfile2"

exit 1

**fi**

**fi**

**comparator3.sh:**

*#!/bin/bash*

**if** ! [[ "$#" = "2" || "$#" = "3" ]]; **then** *#проверка количества аргументов*

exit 1

**fi**

**if** ! [[ -e "$1" && -e "$1" ]]; **then** *#проверка существования файлов*

exit 1

**fi**

**if** ! [[ -f $1 && -f $2 ]]; **then**

exit 1

**fi**

*#проверка, что файл существует и является обычным файлом*

**if** ! [[ -r "$1" || -r "$2" ]]; **then**

exit 1

**fi**

*#проверка, что файл доступен для чтения*

**if** [[ "$#" = "3" ]]; **then**

**if** ! [[ "$3" = "-v" ]]; **then**

exit 1

**fi**

*#Проверка наличия ключа -v*

str1=""

str2=""

count="0"

IFS\_OLD="$IFS"

DONE=false

myfile1="$(mktemp)"

**until** $DONE ;**do**

read -r y || DONE=true

**for** i in $y;

**do**

**if** [[ "$i" =~ ^[+-]?[0-9][0-9]\*[.][0-9]\*$ ]] ; **then**

echo "$i " >> "$myfile1"

count=**$((**$count + 1**))**

**fi**

**if** [[ "$i" =~ ^[+-]?[0-9][0-9]\*$ ]] ; **then**

echo "$i " >> "$myfile1"

count=**$((**$count + 1**))**

**fi**

**done**

**done** < "$1"

**if** [[ $count == "0" ]]; **then**

rm "$myfile1"

exit 1

**fi**

*#Проверка наличия чисел в первом файле*

count="0"

myfile2="$(mktemp)"

DONE=false

**until** $DONE ;**do**

read -r y || DONE=true

**for** i in $y;

**do**

**if** [[ "$i" =~ ^[+-]?[0-9][0-9]\*[.][0-9]\*$ ]] ; **then**

echo "$i " >> "$myfile2"

count=**$((**$count + 1**))**

**fi**

**if** [[ "$i" =~ ^[+-]?[0-9][0-9]\*$ ]] ; **then**

echo "$i " >> "$myfile2"

count=**$((**$count + 1**))**

**fi**

**done**

**done** < "$2"

IFS="$IFS\_OLD"

**if** [[ $count == "0" ]]; **then**

rm "$myfile1" "$myfile2"

exit 1

**fi**

*#Проверка наличия чисел во втором файле*

count="0"

**if** cmp -s "$myfile1" "$myfile2" ; **then**

rm "$myfile1" "$myfile2"

echo "Files are similar"

exit 0

**else**

rm "$myfile1" "$myfile2"

echo "Files are not similar"

exit 1

**fi**

**else**

str1=""

str2=""

count="0"

IFS\_OLD="$IFS"

DONE=false

myfile1="$(mktemp)"

**until** $DONE ;**do**

read -r y || DONE=true

**for** i in $y;

**do**

**if** [[ "$i" =~ ^[+-]?[0-9][0-9]\*[.][0-9]\*$ ]] ; **then**

echo "$i " >> "$myfile1"

count=**$((**$count + 1**))**

**fi**

**if** [[ "$i" =~ ^[+-]?[0-9][0-9]\*$ ]] ; **then**

echo "$i " >> "$myfile1"

count=**$((**$count + 1**))**

**fi**

**done**

**done** < "$1"

**if** [[ $count == "0" ]]; **then**

rm "$myfile1"

exit 1

**fi**

count="0"

myfile2="$(mktemp)"

DONE=false

**until** $DONE ;**do**

read -r y || DONE=true

**for** i in $y;

**do**

**if** [[ "$i" =~ ^[+-]?[0-9][0-9]\*[.][0-9]\*$ ]] ; **then**

echo "$i " >> "$myfile2"

count=**$((**$count + 1**))**

**fi**

**if** [[ "$i" =~ ^[+-]?[0-9][0-9]\*$ ]] ; **then**

echo "$i " >> "$myfile2"

count=**$((**$count + 1**))**

**fi**

**done**

**done** < "$2"

IFS="$IFS\_OLD"

**if** [[ $count == "0" ]]; **then**

rm "$myfile1" "$myfile2"

exit 1

**fi**

count="0"

**if** cmp -s "$myfile1" "$myfile2" ; **then**

rm "$myfile1" "$myfile2"

exit 0

**else**

rm "$myfile1" "$myfile2"

exit 1

**fi**

**fi**

**comparator4.sh:**

*#!/bin/bash*

**if** ! [[ "$#" = "2" || "$#" = "3" ]]; **then** *#проверка количества аргументов*

exit 1

**fi**

**if** ! [[ -e "$1" && -e "$1" ]]; **then** *#проверка существования файлов*

exit 1

**fi**

**if** ! [[ -f $1 && -f $2 ]]; **then**

exit 1

**fi**

*#проверка, что файл существует и является обычным файлом*

**if** ! [[ -r "$1" || -r "$2" ]]; **then**

exit 1

**fi**

*#проверка, что файл доступен для чтения*

**if** [[ "$#" = "3" ]]; **then**

**if** ! [[ "$3" = "-v" ]]; **then**

exit 1

**fi**

*#Проверка наличия ключа -v*

str1=""

str2=""

IFS\_OLD="$IFS"

count="0"

myfile1="$(mktemp)"

DONE=false

**until** $DONE ;

**do**

read -r y || DONE=true

**for** i in $y;

**do**

**if** [[ "$i" =~ ^[+-]?[0-9][0-9]\*[.][0-9]\*$ ]] ; **then**

echo "$i " >> "$myfile1"

count=**$((**$count + 1**))**

**fi**

**if** [[ "$i" =~ ^[+-]?[0-9][0-9]\*[.]?[0-9]\*[eE] \

[-+]?[0-9]\*$ ]] ; **then**

echo "$i " >> "$myfile1"

count=**$((**$count + 1**))**

**fi**

**if** [[ "$i" =~ ^[+-]?[0-9][0-9]\*$ ]] ; **then**

echo "$i " >> "$myfile1"

count=**$((**$count + 1**))**

**fi**

**done**

**done** < "$1"

**if** [[ $count == "0" ]]; **then**

rm "$myfile1"

exit 1

**fi**

*#Проверка наличия чисел в первом файле*

count="0"

myfile2="$(mktemp)"

DONE=false

**until** $DONE ;**do**

read -r y || DONE=true

**for** i in $y;

**do**

**if** [[ "$i" =~ ^[+-]?[0-9][0-9]\*[.][0-9]\*$ ]] ; **then**

echo "$i " >> "$myfile2"

count=**$((**$count + 1**))**

**fi**

**if** [[ "$i" =~ ^[+-]?[0-9][0-9]\*[.]?[0-9]\*[eE] \

[-+]?[0-9]\*$ ]] ; **then**

echo "$i " >> "$myfile2"

count=**$((**$count + 1**))**

**fi**

**if** [[ "$i" =~ ^[+-]?[0-9][0-9]\*$ ]] ; **then**

echo "$i " >> "$myfile2"

count=**$((**$count + 1**))**

**fi**

**done**

**done** < "$2"

IFS="$IFS\_OLD"

**if** [[ $count == "0" ]]; **then**

rm "$myfile1" "$myfile2"

exit 1

**fi**

*#Проверка наличия чисел во втором файле*

count="0"

**if** cmp -s "$myfile1" "$myfile2" ; **then**

rm "$myfile1" "$myfile2"

echo "Files are similar"

exit 0

**else**

rm "$myfile1" "$myfile2"

echo "Files are not similar"

exit 1

**fi**

**else**

str1=""

str2=""

IFS\_OLD="$IFS"

count="0"

myfile1="$(mktemp)"

DONE=false

**until** $DONE ;**do**

read -r y || DONE=true

**for** i in $y;

**do**

**if** [[ "$i" =~ ^[+-]?[0-9][0-9]\*[.][0-9]\*$ ]] ; **then**

echo "$i " >> "$myfile1"

count=**$((**$count + 1**))**

**fi**

**if** [[ "$i" =~ ^[+-]?[0-9][0-9]\*[.]?[0-9]\*[eE] \

[-+]?[0-9]\*$ ]] ; **then**

echo "$i " >> "$myfile1"

count=**$((**$count + 1**))**

**fi**

**if** [[ "$i" =~ ^[+-]?[0-9][0-9]\*$ ]] ; **then**

echo "$i " >> "$myfile1"

count=**$((**$count + 1**))**

**fi**

**done**

**done** < "$1"

**if** [[ $count == "0" ]]; **then**

rm "$myfile1"

exit 1

**fi**

*#Проверка наличия чисел в первом файле*

count="0"

myfile2="$(mktemp)"

DONE=false

**until** $DONE ;**do**

read -r y || DONE=true

**for** i in $y;

**do**

**if** [[ "$i" =~ ^[+-]?[0-9][0-9]\*[.][0-9]\*$ ]] ; **then**

echo "$i " >> "$myfile2"

count=**$((**$count + 1**))**

**fi**

**if** [[ "$i" =~ ^[+-]?[0-9][0-9]\*[.]?[0-9]\*[eE] \

[-+]?[0-9]\*$ ]] ; **then**

echo "$i " >> "$myfile2"

count=**$((**$count + 1**))**

**fi**

**if** [[ "$i" =~ ^[+-]?[0-9][0-9]\*$ ]] ; **then**

echo "$i " >> "$myfile2"

count=**$((**$count + 1**))**

**fi**

**done**

**done** < "$2"

IFS="$IFS\_OLD"

**if** [[ $count == "0" ]]; **then**

rm "$myfile1" "$myfile2"

exit 1

**fi**

*#Проверка наличия чисел во втором файле*

count="0"

**if** cmp -s "$myfile1" "$myfile2" ; **then**

rm "$myfile1" "$myfile2"

exit 0

**else**

rm "$myfile1" "$myfile2"

exit 1

**fi**

**fi**

# Тестирование сценариев командной оболочки

**Тестирование первого варианта скрипта:**

|  |  |  |
| --- | --- | --- |
| **Тесты** | **Ожидаемый результат** | **Полученный результат** |
| Проверка работоспособности скрипта на идентичных файлах | 0 | 0 🡪 Пройден |
| Изменено одно число | 1 | 1 🡪 Пройден |
| Изменено число внутри слова | 0 | 0 🡪 Пройден |
| Изменено два числа | 1 | 1 🡪 Пройден |
| В одном файле нет чисел | 1 | 1 🡪 Пройден |
| В файле только те же числа | 0 | 0 🡪 Пройден |
| Был передан только один аргумент | 1 | 1 🡪 Пройден |
| Были введены не существующие файлы | 1 | 1 🡪 Пройден |

**Тестирование второго варианта скрипта:**

|  |  |  |
| --- | --- | --- |
| **Тесты** | **Ожидаемый результат** | **Полученный результат** |
| Проверка работоспособности скрипта на идентичных файлах | 0 | 0 🡪 Пройден |
| Отсутствует подстрока «string:» | 1 | 1 🡪 Пройден |
| Позиция подстроки не изменилась, но текст изменен | 1 | 1 🡪 Пройден |
| Позиция подстроки и текст изменились | 1 | 1 🡪 Пройден |
| Позиция подстроки не изменилась, но весь текст в одну строчку | 1 | 1 🡪 Пройден |
| Позиция подстроки изменилась, но весь текст в одну строчку | 1 | 1 🡪 Пройден |
| Были введены не существующие файлы | 1 | 1 🡪 Пройден |
| Были введены не существующие файлы | 1 | 1 🡪 Пройден |

**Тестирование третьего варианта скрипта:**

|  |  |  |
| --- | --- | --- |
| **Тесты** | **Ожидаемый результат** | **Полученный результат** |
| Проверка работоспособности скрипта на идентичных файлах | 0 | 0 🡪 Пройден |
| Изменено одно целое число | 1 | 1 🡪 Пройден |
| Изменено одно число с плавающей точкой | 1 | 1 🡪 Пройден |
| Изменены два числа с плавающей точкой | 1 | 1 🡪 Пройден |
| В одном файле нет чисел | 1 | 1 🡪 Пройден |
| В файле только те же числа | 0 | 0 🡪 Пройден |
| Был передан только один аргумент | 1 | 1🡪 Пройден |
| Были введены не существующие файлы | 1 | 1🡪 Пройден |
| Одинаковые значения с плавающей запятой с разными целочисленными значениями между ними | 1 | 1🡪 Пройден |

**Тестирование четвертого варианта скрипта:**

|  |  |  |
| --- | --- | --- |
| **Тесты** | **Ожидаемый результат** | **Полученный результат** |
| Проверка работоспособности скрипта на идентичных файлах | 0 | 0 🡪 Пройден |
| Изменено одно целое число | 1 | 1 🡪 Пройден |
| Изменено одно число с плавающей точкой | 1 | 1 🡪 Пройден |
| В файле только те же числа | 0 | 0 🡪 Пройден |
| Изменены два числа с плавающей точкой(экспоненциальная запись и обычная) | 1 | 1 🡪 Пройден |
| В файле нет чисел | 1 | 1 🡪 Пройден |
| Был передан только один аргумент | 1 | 1 🡪 Пройден |
| Были введены не существующие файлы | 1 | 1 🡪 Пройден |

# Заключение

В ходе выполнения индивидуального итогового задания было реализовано 4 варианта скрипта для сравнения содержимого двух текстовых файлов и скрипты для проверки.

В первом варианте реализовано сравнение последовательностей целых чисел в файлах. В файлах также могут встречаться не только числа, но при сравнении учитываются только они. Сравнение чисел происходит аналогично сравнению текста – 100 и 0100 считаются различными.

Название скрипта: comparator1.sh

Во втором варианте реализовано сравнение текстов в файлах после первого вхождения подстроки «string:». (выполнено сравнение с учётом разницы в пробельных символах и символах окончания строки)

Название скрипта: comparator2.sh

В третьем варианте реализовано сравнение последовательности чисел с плавающей точкой (ЧПТ), записанных не в экспоненциальной форме. В файлах также могут встречаться не только числа, но при сравнении учитываются только они. Сравнение чисел происходит аналогично сравнению текста – 1.0 и 1.00 считаются различными.

Название скрипта: comparator3.sh.

В четвертом варианте реализовано сравнение последовательности чисел с плавающей точкой (ЧПТ), записанных в том числе в экспоненциальной форме. В файлах также могут встречаться не только числа, но при сравнении учитываются только они. Сравнение чисел происходит аналогично сравнению текста – 1.0 и 1.00 считаются различными, числа 1.0 и 1.0e0 – тоже.

Название скрипта: comparator4.sh.

# Список литературы

1. Regular expressions 101

URL: <https://regex101.com/>

1. Regular expressions basics

URL: <https://habr.com/ru/post/545150/>

1. Права доступа

URL: <https://younglinux.info/bash/rwx>

1. Перенаправление ввода-вывода

URL: <https://younglinux.info/bash/input>