

# 作业1

---

## 1. 题目说明

用shell实现以下功能：

1. 传入一个参数：用以遍历该路径下所有文件及目录中文件内容的最长字符串
  1. 该参数传入后，需要做可靠性验证
    - . 若参数为文件，直接对该文件操作即可
  2. 若参数为目录，则遍历该目录
2. 尽量使用函数，程序结构简单明了
3. 最后输出：
  1. 最长字符串的长度
  2. 最长字符串为：
  3. 所在位置（完整的绝对或相对路径）

为了简单起见，字符串是所有用空格，换行符隔开的单元；  
其他所有普通文件，都需要操作；

---

## 2. 代码说明

```
#####
# File Name: getLongestFile.sh
# Author: Lifangqiang
# mail: 576245059@qq.com
# Created Time: 一 8/13 15:26:05 2018
#####

#!/bin/bash
maxString="" #最长的字符串
maxFilePath="" #所包含最长的字符串的文件路径名
maxLength=0 #最长字符串的长度

function getLongestString() {
    file=$1
    echo "Do you want to run this program?[y/n]"
    read validate
```

```

        if [ "$validate" == "y" ] || [ "$validate" == "Y" ] || [ "$validate" ==
"yes" ] || [ "$validate" == "Yes" ] || [ "$validate" == "YES" ]; then #进行
可靠性验证
            if [[ ! -n $file ]]; then #如果getLongestFile.sh后面没有输入任何东西，
那么就getLongestFile .来处理
                file=.
            fi
            for i in `find $file -type f -size -1M`; do #筛选出小于1M的文件的绝对路
径，并且一个路径一个路径的获取。
                filePath=$i
                #echo $filePath
                #string=`cat $filePath`
                #echo $string
                if [[ -f $filePath ]]; then #判断获得的绝对路径是否是文件类型
                    for j in `cat $filePath`; do #以空格和回车为分隔符，分割成一个
一个的字符串，并且一个字符串一个字符串的获取。
                        #echo $j
                        if [[ ${#j} -gt $maxLength ]]; then #如果当前字符串的长度大
于之前最大字符串的长度，那么最长的字符串，最长的字符串的长度，所包含最长的字符串的文件路径
都会更改成当前的。
                            maxString=$j
                            maxFilePath=$filePath
                            maxLength=${#j}
                            #echo $maxString
                            #echo $maxLength
                        fi
                    done
                fi
            done
            if [[ -n $maxFilePath ]]; then
                echo The longest string is: #输出最长的字符串
                echo $maxString
                echo
                echo The file path of longest string is: #输出所包含最长的字符串的文
件路径
                echo $maxFilePath
                echo
                echo The length of longest string is: #输出最长的字符串的长度
                echo $maxLength
            fi
        else
            echo "You are welcome to use this program again."
        fi
    }

getLongestString $1

```

---

### 3. 难点记录

本次作业的难点是：

1. 令我**最头疼**的点是：把文件内容以**空格**和**回车**分隔成一个一个的**字符串**时，如果分隔的**字符串**中，存在**\***或者存在**/**的字符串那么他得到的字符串就是 `ls *` 和 `ls /` 得到的结果。我不管怎么修改脚本，都解决不了这个问题。
  2. 刚开始我由于对**shell**变成不是十分掌握，对基本的**shell**指令的操作不是非常熟悉。我一直在想怎么把文件内容以**空格**和**回车**分隔成一个一个的**字符串**，最终通过和**船友**的讨论最终得出了 `for i in 文件的内容; do` 可以将**文件**内容以**空格**和**回车**分隔成一个一个的**字符串**，最终可以用**if**判断来获取文件中**最长字符串的字符串**、**长度**和**文件路径**。
  3. 如何筛选出**小于1Mb的文件**，通过百度，得出了并通过自己的推敲得出了以下代码： `find $file -type f -size -1M`，此代码筛选出来的是**小于1Mb的文件**，所以得到的都是**文件没有目录**，这样就省出了**递归**的时间，也让**脚本**变得简单。
- 

### 4. 执行结果

1. 在~/other目录的执行结果为：

```
lifaqiang@Lifaqiang_MacBookPro:other$ getLongestFile.sh .
Do you want to run this program?[y/n]
y
█
```

```
1. lifaolang@pi7: ~ (bash)
nts/0//CalendarData_ar_OM!/sun.util.resources.cldr.ar////java.lang.Object[] []getContents/
0//CalendarData_ar_DZ!/sun.util.resources.cldr.ar////java.lang.Object[] []getContents/0//L
ocaleNames_be!/sun.util.resources.cldr.be////java.lang.Object[] []getContents/0//FormatDat
a_dz!/sun.text.resources.cldr.dz////java.lang.Object[] []getContents/0//LocaleNames_mua!/
sun.util.resources.cldr.mua////java.lang.Object[] []getContents/0//LocaleNames_sbp!/sun.ut
il.resources.cldr.sbp////java.lang.Object[] []getContents/0//LocaleNames_cy!/sun.util.reso
urces.cldr.cy////java.lang.Object[] []getContents/0//CalendarData_en_PK!/sun.util.resource
s.cldr.en////java.lang.Object[] []getContents/0//CurrencyNames_ss!/sun.util.resources.cldr
.ss////java.lang.Object[] []getContents/0//FormatData_bn_IN!/sun.text.resources.cldr.bn////
java.lang.Object[] []getContents/0//FormatData_fo!/sun.text.resources.cldr.fo////java.lang
.Object[] []getContents/0//FormatData_mgh!/sun.text.resources.cldr.mgh////java.lang.Object
[] []getContents/0//CurrencyNames_el!/sun.util.resources.cldr.el////java.lang.Object[] []
getContents/0//TimeZoneNames_et!/sun.util.resources.cldr.et////java.lang.Object[] []getCon
tents/0//CalendarData_fr_DJ!/sun.util.resources.cldr.fr////java.lang.Object[] []getConte
nts/0//CurrencyNames_en_PH!/sun.util.resources.cldr.en////java.lang.Object[] []getContents
/0//CurrencyNames_rof!/sun.util.resources.cldr.rof////java.lang.Object[] []getContents/0//
CurrencyNames_ga!/sun.util.resources.cldr.ga////java.lang.Object[] []getContents/0//FormatD
ata_fil!/sun.text.resources.cldr.fil////java.lang.Object[] []getContents/0//CalendarData_g
sw_CH!/sun.util.resources.cldr.gsw////java.lang.Object[] []getContents/0//CalendarData_luo_
KE!/sun.util.resources.cldr.luo////java.lang.Object[] []getContents/0//CurrencyNames_fr_LU/
!/sun.util.resources.cldr.fr////java.lang.Object[] []getContents/0//FormatData_en_Dsrt!/sun
.text.resources.cldr.en////java.lang.Object[] []getContents/0//CurrencyNames_hu!/sun.util.
resources.cldr.hu////java.lang.Object[] []getContents/0//CalendarData_fr_CF!/sun.util.reso
urces.cldr.fr////java.lang.Object[] []getContents/0//CurrencyNames_so_DJ!/sun.util.resourc
es.cldr.so////java.lang.Object[] []getContents/0//CalendarData_saq_KE!/sun.util.resources.
cldr.saq////java.lang.Object[] []getContents/0//FormatData_en_JM!/sun.text.resources.cldr.
en////java.lang.Object[] []getContents/0//CurrencyNames_nn!/sun.util.resources.cldr.nn////
java.lang.Object[] []getContents/0//CalendarData_ga_IE!/sun.util.resources.cldr.ga////java.
lang.Object[] []getContents/0//TimeZoneNames_lt!/sun.util.resources.cldr.lt////java.lang.Ob
ject[] []

The file path of longest string is:
./eclipse-workspace-javase/.metadata/.plugins/org.eclipse.jdt.core/3569978510.index

The length of longest string is:
37278
lifaolang@Lifaolang_MacBookPro:other$
```

2. 在~/other/GitHub目录的执行结果为:

```
lifaolang@Lifaolang_MacBookPro:other$ getLongestFile.sh ./GitHub/
Do you want to run this program?[y/n]
y

```

```
1. lifaqliang@pi7: ~ (bash)
lang.Object[] [] getContents(0//TimeZoneNames_lt!/sun.util.resources.cldr.lt////java.lang.Object[] []
The file path of longest string is:
./eclipse-workspace-javase/.metadata/.plugins/org.eclipse.jdt.core/3569978510.index
The length of longest string is:
37278
lifaqliang@Lifaqliang_MacBookPro:others$ getLongestFile.sh ./GitHub/
Do you want to run this program?[y/n]
y
The longest string is:
VqA00V20d00V={ε 000-y000m0o00,n
T000?rgIihl,0n0S?Gm}!b?030u00u0D0}0z0800A0Kx00ZW0;hS0I0000?-_B=.v_00oI0y9000[^+.P0v0$0000
X&}\00000000 z0h0}0-00/000j}00I-
0v00000Du00I00Uv0000000u00200s~0}[0v0m36]0b0g0dC0600zqn0A0q0NG0M00000?mWj00L00c0500000c-00C000
0x.05000Z&
/O000x0j0,00,0vm@F)X%0i0/b0
b[<U07b0x00V0-0WT0.g1b0(00,000g0};S7000L00E]=0/D00000u000
00q0y0rr00000'RF00 00xe0,0000'(a0I2&0V00}0Q00L00L0$dR00E0cFo'00,0q0;dNADA0}00-0TG0;0h0000u50T0/-_:
0F0x000I000$040o00000vM0000i0!0000G000;b00'00as0?Lw000,*00PyD000m+A0?Jn0f0DwB0000000K0?\'0NSv0
h00000#000;0
0_0&0,0:iYjE0000x0'000}0N00s0ka000000k0Kz0000
0N0'0m000AD000i00
,Xouw0000e00y00cR00000
8p000n00005-00F0i000}a`0(0006X700y)00000
0cc0U0v400000-0xE*0FIF00S'00000A00000Lk00C1+00X0p0000XA0000C^0000C0$Fr0}0R0S00-Z;00j0N}0}HG000
0U0000N000000&+0F0n20x-I00,200-0000RqI0q0w000000j0y0}v>j0XF0R0M0!0C000?0:000k000).00000W00B0L_
0000n000000000J00L0
100p00[X0)F0cx0000-00s00;#00j0i00Q00h0p00'0E00s
The file path of longest string is:
./GitHub//HaiZeI/.git/objects/d3/18d473cc7bfaafefcf9d0abab0ab06e2cc8ca7
The length of longest string is:
1138
lifaqliang@Lifaqliang_MacBookPro:others$
```

都是乱码，我也很无奈！

## 5.改进方法

我的改进方法是：

1. 文件范围**可执行文件**和**不可执行文件**，而**可执行文件**用cat输出内容输出的是**乱码**。因此在查找最长字符串时得到的字符串时三万多行的**乱码**，根本不具有代表性！如何改进成一个获得不含**乱码**的文件中**最长的字符串**，这样比较有**实用性**。
2. 如何将存在\*或者存在/\*的字符串以原来的**字符串**输出，即特殊的**字符串**不按特殊处理。

