



浙江大学

MATHEMATICAL REPORT

MATH SOFTWARE

Simple Try of Shell

Author: Haowei Zheng

Studnet Id: 3200104204

Date: 2022 年 6 月 28 日

1 问题确立

课本在 **P43** 中的 TRY IT OUT 中简述了如何在 Shell 中创建一个简单的函数，代码如下：

Listing 1: Try It Out

```
1  #!/bin/sh
2  foo() {
3      echo "Function foo is executing"
4  }
5  echo "script starting"
6  foo
7  echo "script ended"
8  exit 0
```

并且在后续测试中发现，在传值进入新的函数中时，`$* $1 $2 ...` 的值会改变，直到函数执行完毕后再变回原来的值。但是课本中并没有讲到递归的实现情况。为此，我用 Shell 编写了一个汉诺塔的程序。分析其中对应的 `$* $1 $2 ...` 的值的变化情况。

2 程序实现

首先考虑输入的整数 n ，需要进行判断是否为纯数字并且为正整数。为此，我们考虑利用 `expr` 操作来进行判断。我们将 n 读入到 `$x` 当中，并利用 `expr $x+0` 判断是否为整数。如果返回值不为 0，说明加法成功，即 n 为整数。同时，为了不让错误信息输出到标准流当中，我们调用 `&>/dev/null` 将错误信息丢弃。

接下来判断是否为负数。直接拿 n 与 0 进行比较大可即可，判断命令为 `[$x -le 0]`。

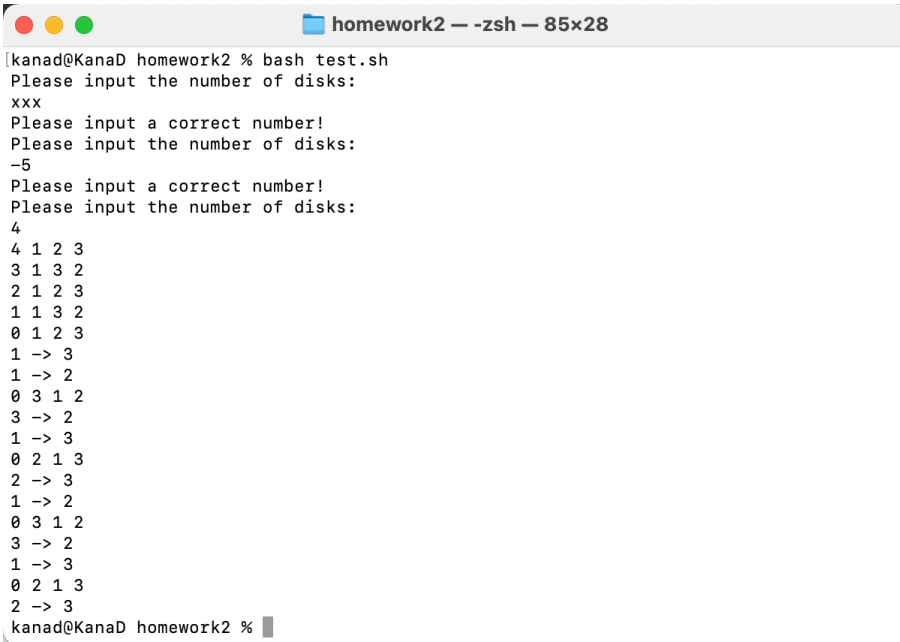
最后就是进行递归并且输出每次调用函数对应的 `$* $1 $2 ...`，观察变化。对应的输出指令为 `echo $*`。

最终代码如下：

Listing 2: code

```
1      #!/bin/sh
2      work(){
3          echo $*
4          if [ $1 = 0 ]
5          then
6              echo "$2 -> $4"
7          else
8              x=$(expr $1 - 1)
9              work $x $2 $4 $3
10             echo "$2 -> $4"
11             work $x $3 $2 $4
12         fi
13     }
14     while true
15     do
16         echo "Please input the number of disks:"
17         read x
18         expr $x + 0 &>/dev/null
19         if [ $? = 0 ]
20         then
21             if [ $x -le 0 ]
22             then echo "Please input a correct number!"
23             else
24                 work $x 1 2 3
25                 exit 0
26             fi
27         else
28             echo "Please input a correct number!"
29         fi
30     done
31     exit 0
```

并且得到以下输出:



```
kanad@KanaD homework2 % bash test.sh
Please input the number of disks:
xxx
Please input a correct number!
Please input the number of disks:
-5
Please input a correct number!
Please input the number of disks:
4
4 1 2 3
3 1 3 2
2 1 2 3
1 1 3 2
0 1 2 3
1 -> 3
1 -> 2
0 3 1 2
3 -> 2
1 -> 3
0 2 1 3
2 -> 3
1 -> 2
0 3 1 2
3 -> 2
1 -> 3
0 2 1 3
2 -> 3
kanad@KanaD homework2 %
```

图 1: result

由此可见, 在递归的情况下, $\$*$ $\$1$ $\$2 \dots$ 的值也会递归变化并逐级变回来。