

GF3002

脚本语言

刘志伟

刘志伟

Lua函数简单应用

余庆祥

2220631136

BIH

2022/10/9

**内页写作格式**

**一、实验名称：**要用最简练的语言反映实验的内容。

**二、实验日期：**写明做实验的具体年、月、日及组別。

**三、实验目的：**使用简洁的文字或关键字来敘述，是以怎样的目的作此实验的。

**四、实验环境：**实验的操作系统和软件等。

**五、实验的步骤和方法：**这是实验报告极其重要的内容。这部分要写明经过哪几个步骤。

**六、数据记录和计算：**指从实验中测到的数据以及计算结果。

**七、实验结果或结论：**即根据实验过程中所见到的现象和测得的数据，得出结论。

**八、备注或说明**：可写上实验成功或失败的原因，实验后的心得体会、建议等。

|  |  |
| --- | --- |
| **1、实验名称** | Lua函数简单应用 |
| **2、实验日期** | **2022.10.09** |
| **3、实验目的** | |
| （1）理解lua函数的概念、返回值的应用；  （2）掌握lua函数的定义和调用；  （3）掌握局部函数的定义和调用。 | |
| **4、实验环境** | |
| 1、win7以上兼容系统，lua解释器 | |
| **5、实验的步骤和方法** | |
| 1. 定义生成RandNum函数 2. 构造石头剪刀布table、玩家结果table 3. 定义game函数，先判断玩家输入的数字是否符合1~3的范围内，如不符合给出提示，用输入的玩家数字减去生成的随机数所得数判断输赢，再计算玩家赢的局数 4. 定义main函数，利用输入局数控制循环次数，在main函数中调用RandNum函数和game函数实现剪刀石头布游戏，最后统计玩家及计算机赢的次数 5. 定义获取随机数函数 6. 定义需要到的局部变量 7. 开始循环打印提示信息和获取用户输入 8. 判断是否是非正常输入与退出命令 9. 判断哪方胜负与平局和打印相关信息 10. 异常输入的处理 | |
| **6、数据记录和计算** | |
| 1. 定义获取随机数函数代码：   local function getRandom()  local random = math.random() \* 1000000  return (random - (random % 1)) % 3 + 1  end   1. 定义需要到的局部变量代码：   local player  local player\_win = 0  local computer  local computer\_win = 0  local draw = 0  local invalid\_input = 0  local count = 0  local numberToString = {"scissors", "stone", "paper"}   1. 开始循环打印提示信息和获取用户输入代码：   while true do  print("-----------------------------------------------")  print("Please input a number 1~4(1.scissors, 2.stone, 3.paper, 4.quit): ")  player = io.read()   1. 判断是否是非正常输入与退出命令代码：   if player == "1" or player == "2" or player == "3" or player == "4" then  player = tonumber(player)  if player == 4 then  print("play " .. count .. " times, computer win:" .. computer\_win  .. ", you win:" .. player\_win .. ", draw:" .. draw ..  ", invalid input:" .. invalid\_input) -- 结束时输出的内容  break  end   1. 判断哪方胜负与平局和打印相关信息代码：   computer = getRandom()  print(computer)  if player == computer then -- 平局时  draw = draw + 1  print("draw")  elseif player == 1 then -- 玩家输入1时  if computer == 2 then  computer\_win = computer\_win + 1  print("computer win")  elseif computer == 3 then  player\_win = player\_win + 1  print("you win")  end  elseif player == 2 then -- 玩家输入2时  if computer == 3 then  computer\_win = computer\_win + 1  print("computer win")  elseif computer == 1 then  player\_win = player\_win + 1  print("you win")  end  elseif player == 3 then -- 玩家输入3时  if computer == 1 then  computer\_win = computer\_win + 1  print("computer win")  elseif computer == 2 then  player\_win = player\_win + 1  print("you win")  end  end  print("")  print("you:" .. numberToString[player])  print("computer:" .. numberToString[computer])   1. 异常输入的处理代码：   else  invalid\_input = invalid\_input + 1 -- 异常输入时  print("Invalid number")  end  所有代码：  local function getRandom()      local random = math.random() \* 1000000      return (random - (random % 1)) % 3 + 1  end  local function game()      local player      local player\_win = 0      local computer      local computer\_win = 0      local draw = 0      local invalid\_input = 0      local count = 0      local numberToString = {"scissors", "stone", "paper"}      while true do          print("-----------------------------------------------")          print("Please input a number 1~4(1.scissors, 2.stone, 3.paper, 4.quit): ")          player = io.read()          if player == "1" or player == "2" or player == "3" or player == "4" then              player = tonumber(player)              if player == 4 then                  print("play " .. count .. " times, computer win:" .. computer\_win                          .. ", you win:" .. player\_win .. ", draw:" .. draw ..                          ", invalid input:" .. invalid\_input) -- 结束时输出的内容                  break              end              computer = getRandom()              if player == computer then -- 平局时                  draw = draw + 1                  print("draw")              elseif player == 1 then -- 玩家输入1时                  if computer == 2 then                      computer\_win = computer\_win + 1                      print("computer win")                  elseif computer == 3 then                      player\_win = player\_win + 1                      print("you win")                  end              elseif player == 2 then -- 玩家输入2时                  if computer == 3 then                      computer\_win = computer\_win + 1                      print("computer win")                  elseif computer == 1 then                      player\_win = player\_win + 1                      print("you win")                  end              elseif player == 3 then -- 玩家输入3时                  if computer == 1 then                      computer\_win = computer\_win + 1                      print("computer win")                  elseif computer == 2 then                      player\_win = player\_win + 1                      print("you win")                  end              end              print("")              print("you:" .. numberToString[player])              print("computer:" .. numberToString[computer])          else              invalid\_input = invalid\_input + 1 -- 异常输入时              print("Invalid number")          end          count = count + 1 -- 每次输入计数一次      end  end  local function main()      math.randomseed(os.time())      game()  end  main() | |
| **7、实验结果或结论（**实验结果怎么样？你从这个实验你学会了什么？得出了什么结论？） | |
| **进入游戏后，根据提示输入1-3进行游戏，会与系统生成的随机数进行比较，根据预先的判断判定谁赢谁输; 输入4则退出游戏; 输入其它内容则继续计数且不进行比较。** | |
| **8、备注或说明**  **）** | |
| Lua没有switch语句，就需要靠比较多的if语句来实线；  Lua没有continue语句，所以在循环时，需要考虑到什么情况下不执行后面的代码；  石头剪刀布的判断是把所有可能性列出来，应该还有更优雅的写法，有待优化。 | |