# Python程序设计

# 亲例: 猜单词游戏程序



张华 WHU

## 猜单词游戏

## ■游戏规则

猜单词游戏是一种简单的游戏,计算机从指定单词列表中抽取一个单词,通过算法,把单词的字母顺序打乱,然后输出给玩家猜测。玩家根据乱序的字母,组合猜测输入正确的单词。计算机确定是否猜测正确。

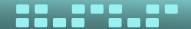


## 猜单词游戏程序

#### ■ 设计思路

使用元组或列表构建待猜测的英文单词库列表WORDS,使用random模块的choice函数从单词的元组中随机抽取一个英文单词word。然后把该英文单词的字母乱序排列(方法是每次随机抽取一个位置的字符放入乱序的jumble字符串中,并从原word中删除该字符)。

游戏一开始先显示乱序后的字符串 jumble,并提示用户输入猜测的结果,如果错误,提示继续输入,直至输入正确。猜对之后,可以询问是否继续游戏。游戏也可以通过Ctrl+C强制中断运行。



#### COMPUTER PROGRAMMING

## 猜单词游戏程序

#### ■ 实现

```
#模块: word_jumble.py
import random

#建立英文单词库,以列表的形式表示

WORDS = ['apple','pear','banana','cherry','good','better','best',
'python','while','tuple','dictionary','jumble','difficult',
'aesthetic','stereotype','civilization','anniversary']

print("欢迎参加猜单词游戏!\n请把乱序后的字母组成一个单词\n")
```

## 猜单词游戏程序

#### ■ 实现

```
isContinue = "Y"
while isContinue in ("Y", "y"):
   #随机挑选一个单词
   word = random.choice(WORDS)
   answer = word #保存答案
   #将选出的单词进行乱序
   jumble = ""
   for i in word:
       #随机抽取一个位置的字符放入乱序 jumble 中,并从原word中删除该字符
       position = random.randrange(len(word))
       jumble += word[position]
       word = word[:position] + word[(position+1):]
   #输出乱序后的单词
   print("乱序后的单词: ", jumble)
```

#### COMPUTER PROGRAMMING

## 猜单词游戏程序

#### ■ 实现

```
#接受玩家猜的单词
guess = input("请输入您猜测的结果: ")
while guess != answer:
    guess = input("结果不对,请重新猜测: ")

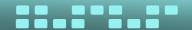
print("恭喜您,猜对了! ")

#询问是否重复游戏
isContinue = input("是否继续(Y/N)?")

print("谢谢参与,欢迎下次再玩!")
```

欢迎参加猜单词游戏! 请把乱序后的字母组成一个单词

乱序后的单词: elapp 请输入您猜测的结果: apple 恭不知的结果: apple 表否继续(Y/N)?y 乱序后继续的单词: epltu 请补入您猜对了! 起序后您猜对了! 是不好,请对了! 是所入您有测的结果: ooooooo 结果不对,请重新猜测: python 结果不对,请重新猜测: python 结果不对,请对了! 是一个人的?n 数少的,欢迎下次再玩!



## 猜单词游戏程序

## ■扩展程序功能

- \* 从文件中读入单词列表
- \* 记录游戏玩家的得分
- \* 记录玩家猜对单词需要的平均时间和次数
- \*根据单词的长度选择出现的顺序,即不断增加难度
- \* 进一步根据单词的生僻程度设置难度
  - >需要统计单词的生僻值
- \* 对单词进行分类,给玩家适当提示

