



UNIVERSIDADE FEDERAL DO PARANÁ
CURSO DE MATEMÁTICA

Ana Flávia Sturaro Calegari
Danyelle Horobinski
Jéssica Gomes Furtado

JOGO LUDO EM PYTHON

CURITIBA
2018

GRR20185651
GRR20185654
GRR20185640

LUDO EM PYTHON

Relatório apresentado à disciplina
Fundamentos de Programação de
Computação do Curso de Matemática
da Universidade Federal do Paraná.

Orientador: Prof. Jackson Antônio do
Prado Lima

CURITIBA
2018

INTRODUÇÃO

Usamos python, uma linguagem de programação criada por Guido Van Rossum, para fazer um jogo ludo.

O jogo é composto por um tabuleiro, contendo as cores verde, amarela, azul e vermelha. É jogado por quatro jogadores, um o usuário e os outros três serão o computador, utilizaremos um dado de seis lados que nos informará o quanto lhe é permitido caminhar no jogo.

2. OBJETIVOS

2.1 OBJETIVO GERAL

O objetivo é obter o jogo ludo, que tenha quatro jogadores, o usuário jogará dando um click na tela, que fará o papel do dado, dando um número aleatório. Sua missão é chegar primeiro a caixa central percorrendo todo o tabuleiro. Ganha o que primeiro conduzir as sua peças até o centro.

DESENVOLVIMENTO

CÓDIGO LUDO:

Usamos o módulo turtle para fazer o tabuleiro e os jogadores.
Turtle é um módulo em Python que permite fazer desenhos na tela com comandos simples.

Começamos nosso código importando o módulo Turtle.

Usamos o módulo para a construção dos nossos jogadores, com os seguintes comandos:

- Primeiro criamos o nome da nossa variável e a inserimos no módulo turtle.
- Depois definimos seu formato com o comando “shape” e em seguida definimos sua cor com o comando “color”.
- Em seguida usamos o comando “speed” para definir a velocidade em que nossos jogadores irão se movimentar.
- Usamos o comando “up” que é para levantar nosso jogador e levar ele para determinada coordenada com o comando “goto”
- E por ultimo usamos o comando “left” para rotacionar 90 graus nosso jogador. Assim foi feito com as demais cores.

```
import turtle

#####criando a tartaruga azul#####

turtle_blue=turtle.Turtle()
turtle_blue.shape("turtle")
turtle_blue.color("black","blue")
turtle_blue.speed(25)
turtle_blue.up()
turtle_blue.goto(120,-200)
turtle_blue.left(90)
```

- Usamos os mesmos comandos dos jogadores para a criação do nosso dado.

```
#####criando a tartaruga dado#####

turtle_dado=turtle.Turtle()
turtle_dado.shape("classic")
turtle_dado.speed(25)
turtle_dado.up()
```

Aqui começamos a criar nosso tabuleiro, com os seguintes comandos:

- Usamos o nome “tabuleiro” para a variável que utilizamos para desenhar nosso tabuleiro no Turtle.
- Em seguida escolhemos a forma da caneta, sua velocidade e sua cor.
- Logo após utilizamos o comando “begin_fill”, que serve para começar a preencher o desenho com a cor desejada.

```
#####criando o tabuleiro#####

tabuleiro = turtle.Turtle() #criando um "nome" para o modulo turtle

tabuleiro.shape("turtle") #escolhe a caneta
tabuleiro.speed(11) #velocidade da tartaruga
tabuleiro.color("blue","blue") #cores do poligono
tabuleiro.begin_fill() #inicia o preenchimento do poligono
```

- Em seguida, utilizamos os comandos “forward” e “right” que servem para andar com a caneta para frente e rotacionar a caneta, respectivamente.

```
tabuleiro.forward(200) #200 unidades para frente
tabuleiro.right(90) #90 graus para a direita
```

- Utilizamos a função “end_fill” para terminar de preencher nossa base e flecha de determinada cor. Assim fizemos para todos os quatro jogadores.

```
tabuleiro.end_fill() #Termina o preenchimento do poligono
```

- Aqui definimos o tamanho do nosso tabuleiro e usamos o comando “up” para levantar a caneta e o comando “down” para abaixar a caneta e começar a desenhar.

```
#delimitando o tabuleiro

tabuleiro.up() #levanta a caneta
tabuleiro.right(90)
tabuleiro.forward(240)
tabuleiro.left(90)
tabuleiro.forward(240)
tabuleiro.down() #abaixa a caneta

tabuleiro.color("black") #mudando a cor da caneta

for i in range(0,4):
    tabuleiro.left(90)
    tabuleiro.forward(600)
```

- Em seguida usamos os mesmos comandos citados anteriormente para a criação das casas do nosso tabuleiro.

```
#fazendo os quadradinhos

tabuleiro.color("black")

#divisão azul e vermelho

tabuleiro.up()
tabuleiro.forward(40)
tabuleiro.left(90)
tabuleiro.forward(40)
tabuleiro.down()
```

Para o desenho dos valores do dado usamos os seguintes comandos:

- “up”, “shape”, “color”, “right”, “forward”, “stamp”, que serve para carimbar determinado desenho e “back” para voltar com a caneta para determinado lugar.

```
#=====montando os valores=====#
turtle_dado.up()
turtle_dado.shape("circle")
turtle_dado.color("black","orange")
turtle_dado.back(20)
turtle_dado.right(90)
turtle_dado.forward(20)
turtle_dado.stamp()
for x in range(0,3):
    turtle_dado.forward(80)
    turtle_dado.right(90)
    turtle_dado.stamp()
```

Em seguida fizemos uma função para mostrar no dado que está na tela qual o valor que cada jogador tirou.

Fizemos isso para os valores de 1 a 6.

```
#-----funções dado-----#
def dado_turtle(dado,vez):
    turtle_dado.speed(2)
    if dado==6:
        if vez=="Blue":
            turtle_dado.color("black","blue")
        elif vez=="Yellow":
            turtle_dado.color("black","yellow")
        elif vez=="Green":
            turtle_dado.color("black","green")
        if vez=="Red":
            turtle_dado.color("black","red")
    turtle_dado.shape("turtle")
    turtle_dado.goto(460,-20)
```

- Em seguida importamos o módulo random, que é um módulo para gerar números aleatórios
- E depois criamos uma lista para nos auxiliar nas delimitações do nosso tabuleiro.

```
from random import* #importando random (aleatório)

#-----criando uma lista tabuleiro-----#

lista_tabuleiro=["inicio"]

for x in range(0,62): #caminho para percorrer ("tabuleiro")
    x="~"
    lista_tabuleiro.append(x)
```

- Logo após a criação da nossa lista auxiliar criamos uma lista com a posição de todas as esquinas, para que nossos jogadores pudessem virar nas esquinas e não sair do nosso tabuleiro.

```
#####criando uma lista das esquinas#####

lista_esquina=["inicio"]
for x in range(0,62):
    x="^"
    lista_esquina.append(x)

lista_esquina[3]="E"
lista_esquina[9]="E"
lista_esquina[15]="E"
lista_esquina[17]="E"
lista_esquina[23]="E"
lista_esquina[29]="E"
lista_esquina[31]="E"
lista_esquina[37]="E"
lista_esquina[43]="E"
lista_esquina[45]="E"
lista_esquina[51]="E"
lista_esquina[56]="E"
lista_esquina[57]="E"
```

- Aqui criamos uma função para que nosso jogador pudesse andar pelo tabuleiro.
- E criamos condições com a lista auxiliar de esquinas para os usuários desviarem delas.
- Fizemos isso com todas as 4 cores.

```
#####funções para os bonequinhos andarem#####

def bonequinho_azul(b,andar):
    turtle_blue.speed(2)
    turtle_blue.shape("turtle")
    turtle_blue.color("Black","Blue")
    lista_auxiliar=[]
    for x in range(b+1,andar+b+1):
        lista_auxiliar.append(lista_esquina[x])

    if b==0:
        turtle_blue.goto(220,-100) #inicio azul
        turtle_blue.forward(andar*40)
```

Criamos funções para o início do jogo, para cada cor, dependendo da cor que for escolhida pelo usuário.

```
#####inicindo o jogo#####

def apertou_azul():
    jogador="Blue"
    jogo(jogador)

def apertou_amarelo():
    jogador="Yellow"
    jogo(jogador)

def apertou_verde():
    jogador="Green"
    jogo(jogador)

def apertou_vermelho():
    jogador="Red"
    jogo(jogador)
```

Aqui criamos a função do jogo, e que o jogo só começa com o jogador tirando 6 no dado.

```
#####vamos à partida#####

def jogo(jogador):

    vez="Blue"
    nao_entrarB=0 #para que na primeira rodada os bonequinhos só comecem a avançar quando tirarem seis no dado
    nao_entrarR=0 #precisa de um para cada cor
    nao_entrarG=0
    nao_entrarY=0

    b=0 #contador azul (primeiro bonequinho) caminhada
    r=0 #contador vermelho (primeiro bonequinho) caminhada
    g=0 #contador verde (primeiro bonequinho) caminhada
    y=0 #contador amarelo (primeiro bonequinho) caminhada
```


Criamos essa parte para identificar o ganhador e mostrar na tela.

```
#####identificar o vencedor#####  
  
if vez=="Yellow":  
    vencedor=Tk()  
    vencedor_texto=Label(vencedor,text="AZUL VENCEU!!!",fg="blue")  
    vencedor_texto.place(x=70,y=65)  
    if jogador=="Blue":  
        vencedor_texto2=Label(vencedor,text="PARABÉNS! VOCÊ VENCEU!")  
        vencedor_texto2.place(x=30,y=100)  
    else:  
        vencedor_texto2=Label(vencedor,text="SINTO MUITO! VOCÊ PERDEU!")  
        vencedor_texto2.place(x=50,y=100)
```

- Por último usamos o módulo tkinter para criar uma interface de interação com o usuário
- E chamamos nossa função para começar o jogo.

```
from tkinter import * #importando o módulo de interfaces  
  
def começar_jogo():  
  
    teste=Tk()  
  
    botao_azul=Button(teste,width=20,text="AZUL",fg="blue",command=apertou_azul)  
    botao_azul.place(x=60,y=100)  
  
    botao_amarelo=Button(teste,width=20,text="AMARELO",fg="yellow",command=apertou_amarelo)  
    botao_amarelo.place(x=60,y=150)  
  
    botao_verde=Button(teste,width=20,text="VERDE",fg="green",command=apertou_verde)  
    botao_verde.place(x=60,y=200)  
  
    botao_vermelho=Button(teste,width=20,text="VERMELHO",fg="red",command=apertou_vermelho)  
    botao_vermelho.place(x=60,y=250)  
  
    texto1=Label(teste, text="Bem-Vindo, ao LUDO PYTHON!!!")  
    texto1.place(x=50,y=10)  
  
    texto2=Label(teste,text="Para iniciar a diversão, escolha seu jogador:")  
    texto2.place(x=20,y=40)  
  
    teste.geometry("300x300+200+200")  
    teste.mainloop()  
  
começar_jogo() #chamando a função para iniciar o jogo
```

Conclusão

Nosso maior desafio foi a interação com o usuário. Mas conseguimos fazer um ótimo jogo com o que foi aprendido em sala.