

BOUNCING BALL TEAM - 1

Date:22-04-2022

TEAM MEMBERS

20B01A1269- K.Navya Sri IT

20B01A1272- K.Geetha IT

20B01A1271- K.Kavya IT

20B01A0266- J.Lakshmi Sreeja EEE

20B01A0261- K.Mounika Lakshmi EEE

20B01A0265- K.Asha Nagavalli EEE

"Bouncing Ball" is fun game . It is a single player game. The goal of the game is to clear the balls in the playing field by hitting them with a ball from the paddle. The game ends when ball touches the ground.

Description

The game starts when user press 'Enter' button. The ball on the paddle hits the balls at the top of the Game board and it bounce back to bottom of the screen. The ball should hit the paddle to bounce again. The game ends when ball touches the ground without touching the paddle.

Project Flow

Day 1

- Brainstorming ideas on problem description

Day 2

- Worked on the modules of tkinter and created canvas

Day 3

- Built the logic and GUI

Day 4

- Worked on the LaTeX, gitlab and Debugging

Day 5

- Presentation

TEAM CONTRIBUTION

- 20B01A1269 - Building code ,creating project in GitLab ,Preparing the LaTeX
- 20B01A1271 - Collecting the requirements,Building code
- 20B01A1272 - Collecting information on code,helping in building code
- 20B01A0266 - Helping in presentation, building the code
- 20B01A0265 - Preparing LaTeX file,collecting requirements
- 20B01A0261 - Building the code,latex

Challenges

- Creating GUI window
- We faced difficulty to move the paddle in certain direction
- We faced a hurdle in bouncing the ball in bottom direction

OVERCOMING CHALLENGES

- Using tkinter package we create the GUI widow
- Solved the problem by using modular approach
- By dividing the problems into parts and working on each module separately

Learnings

- Tkinter
- Random module
- Time module
- Canvas

Tech Stack

- Programming Languages used: Python
- Tools used: idle
- Packages used:tkinter ,time, random

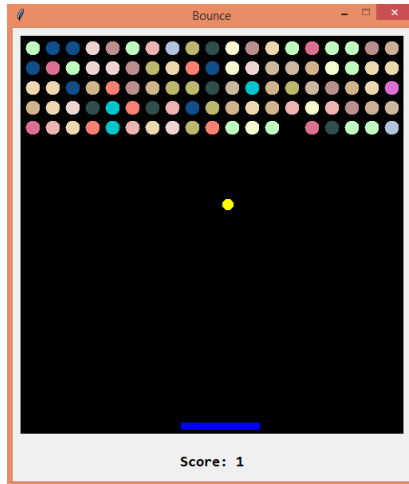
CODE

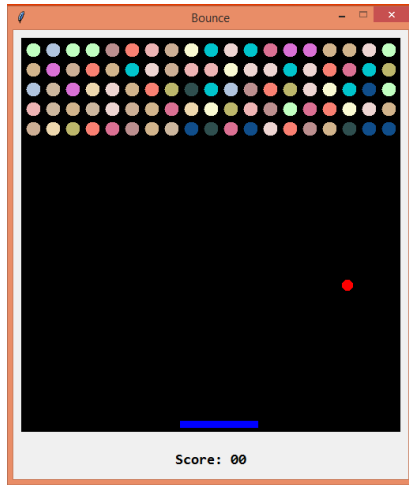
```
1 from tkinter import *
2 import time
3 import random
4
5 root = Tk()
6 root.title("Bounce")
7 root.geometry("500x570")
8 root.resizable(0, 0)
9 root.wm_attributes("-topmost", 1)
10 canvas = Canvas(root, width=500, height=500, bd=0, highlightthickness=0, bg="Black")
11 canvas.pack(padx=10, pady=10)
12 score = Label(height=50, width=80, text="Score: 00", font="Consolas 14 bold")
13 score.pack(side="left")
14 root.update()
15
16
17 class Ball:
18     def __init__(self, canvas, color, paddle, bricks, score):
19         self.bricks = bricks
20         self.canvas = canvas
21         self.paddle = paddle
22         self.score = score
23         self.bottom_hit = False
24         self.hit = 0
25         self.id = canvas.create_oval(10, 10, 25, 25, fill=color, width=1)
26         self.canvas.move(self.id, 230, 461)
27         start = [4, 3.8, 3.6, 3.4, 3.2, 3, 2.8, 2.6]
```



Output Screenshot









Code Statistics

- Number of Lines of Code :195
- Number of Functions :11
- Number of classes : 3

REPOSITORY

`https://gitlab.com/wise_batch1/
batch1_bouncing_ball_wise.git`

Let us see the Demo

THANK YOU