

Project Report

The program meets all the objectives

1) Collision Detection:

- Implement at least two moving shapes
 - There is a the player ball and three circular objects which are machine opponents
- Write logic to detect collision between the shapes
 - There are multiple collision detection calculations implemented in the **App:updatePosition** method
 - Ball Collides with walls
 - Circular obstacle A collides with walls
 - Circular obstacle B collides with walls
 - Circular obstacle C collides with walls
 - Ball Collides with Rect obstacles
 - Circular obstacle A collides with Rect obstacles
 - Circular obstacle B collides with Rect obstacles
 - Circular obstacle C collides with Rect obstacles
 - Ball Collides with Circular obstacle A
 - Ball Collides with Circular obstacle B
 - Ball Collides with Circular obstacle C
 - Circular obstacle A Collides with Circular obstacle B
 - Circular obstacle A Collides with Circular obstacle C
 - Circular obstacle B Collides with Circular obstacle C
 - Trigger a response upon collision
 - Whenever the player ball collides with a circular obstacle the color of the player ball changes. Additionally, the score goes up by 1 which is displayed.

2) Button and Mouse Events:

- Add at least two interactive elements:
- Buttons:
 - There is menu system to access help, credits, exit which is activated by mouse click
 - There are three button to load a different map and start the game
 - There is a back button to exit the game and return to the navigation window
- Mouse:
 - There is a cheat code where you can drag the player ball with the mouse and avoid Newton's laws of physics.
- Keys:

- There is motion of the player ball which can be impacted by using the arrow keys.

3) Abstract Classes:

- Design an Abstract class to represent a core component of the game
 - All Obstacles that a player ball can collide with inherit from the AbstractObstacle class
- Concrete Subclasses
 - The CircularObstacle and the RectObstacle inherit from the Abstract class are concrete

4) Gui Design:

- Create A User Friendly Window
 - There are two scenes in the program: The navigation scene, where individuals can pick a map and get help and the gamescene where individuals can play the game.
 - The game scene animates the game and displays a running timer and score in the top pane