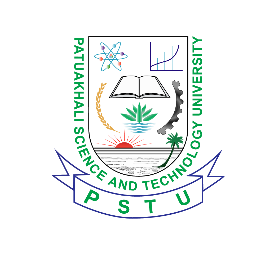
****

**PATUAKHALI SCAIENCE AND TECHNOLOGY UNIVERSITY**

***COURSE CODE: CCE-112***

***Assignment title:*** A POLISHED BRICK BREAKER GAME

***Assignment No: 12***

**SUBMITTED TO*:***

**Prof. Dr. Md Samsuzzaman Sobuz**

**Department of Computer Science and Technology**

**Faculty of Computer Science and Engineering**

**SUBMITTED BY:**

**Name: Md. Mehedi Hasan**

**ID: 2102016**

**Reg. No: 10143**

***F*aculty of Computer Science and Engineering**

**SUBMISSION DATE:**

**12/04/2023**

**REVIEW QUESTIONS**

1. **How does the program know when the Tennis Ball sprite has gotten past the Paddle sprite?**

Answer: To track the position of the Tennis Ball sprite in this program, an infinite loop is used to continuously monitor its Y-coordinate. If the Y-coordinate of the Tennis Ball sprite drops below a certain boundary, it means that the player has missed the ball and the game ends, resulting in the loss of a life. To implement this functionality, an "if" statement is used to check if the Y-coordinate is below the boundary. If it is, the game ends and the Tennis Ball sprite is reset to its initial position at the top of the screen. The player's lives are also decremented, and if the player has no lives left, the game over message is broadcasted and the program ends. By tracking the position of the Tennis Ball sprite in this way, a simple but functional game can be created in Scratch.

1. **Which block creates clones of a sprite?**

answer: In Scratch, the "create clone of [myself]" block is used to create clones of a sprite. This block can be found in the "Control" category of the Scratch editor, and can be used to make a sprite create one or more clones of itself. When this block is executed, a new clone sprite is created that is identical to the original sprite, including its position, appearance, and scripts. The new clone sprite is added to the stage, and can be moved and controlled independently from the original sprite. This block is useful for creating multiple instances of a sprite, which can be used to create more complex animations and games in Scratch. By using the "create clone of [myself]" block, it's possible to make sprites that can interact with each other and the player, leading to more engaging and interactive projects.

1. **Which block has the code that clones run when they are created?**

Answer:

In Scratch, the "when I start as a clone" block contains the code that clones run when they are created. This block is a hat block that can be found in the "Events" category of the Scratch editor. When a new clone sprite is created using the "create clone of [myself]" block, the code inside the "when I start as a clone" block is executed for the clone sprite. This block allows the programmer to specify the initial behavior of the clone sprite, including its position, movement, appearance, and any other scripts that it should execute. By using the "when I start as a clone" block, it's possible to create clones that behave differently from the original sprite, leading to more complex and dynamic projects in Scratch.

1. **What are the three rotation styles?**

Answer:

In Scratch, there are three rotation styles that can be applied to sprites: "normal", "left-right", and "don't rotate". These rotation styles can be found in the "Motion" category of the Scratch editor. The "normal" rotation style is the default and allows the sprite to rotate freely in any direction. The "left-right" rotation style only allows the sprite to rotate around its Y-axis, so it always faces either left or right. This rotation style is often used for characters that move horizontally, such as a platformer game. The "don't rotate" rotation style disables rotation entirely, meaning that the sprite always faces the same direction regardless of its movement. This rotation style is useful for sprites that should remain fixed in a certain position, such as a background or a user interface element. By using the different rotation styles, it's possible to create sprites with unique movements and behaviors in Scratch.

5**. Why do the You Win and Game Over sprites hide themselves**

**after you click the green flag?**

Answer:

In Scratch, the "You Win" and "Game Over" sprites may hide themselves after the green flag is clicked in order to reset the game to its initial state. By hiding these sprites, they will not appear on the stage when the game is restarted, ensuring that the player starts with a clean slate. Hiding these sprites can also be used to create a sense of anticipation and excitement for the player, as they won't know if they've won or lost until they reach the end of the game. Additionally, hiding these sprites can help to keep the stage clutter-free, making it easier for the player to focus on the gameplay. By hiding the "You Win" and "Game Over" sprites after the green flag is clicked, the programmer can create a more streamlined and polished game experience for the player.

6**. What does the wait until block do?**

Answer:

In Scratch, the "wait until" block is a useful alternative to using a forever loop with an if condition. This block takes a conditional parameter, and the code within it won't be executed until the condition is met. This allows the programmer to create more efficient and streamlined code, as they can avoid using a loop that runs continuously and wastes resources. By using the "wait until" block, the program can wait for a specific condition to be met, such as the player pressing a certain key or a certain sprite moving to a certain position, before executing the code. This block is particularly useful in games and interactive projects, where the program needs to respond to user input or events in a timely and accurate manner. By using the "wait until" block, programmers can create more responsive and interactive projects in Scratch.