Report

CS32 Project 3

Kareem Dibs

1. A description of the control flow for the interaction of a player avatar and a bank square. Where in the code is the co-location of the two objects detected, and what happens from that point until the interaction is finished? Which functions of which objects are called and what do they do during the handling of this situation?

When the player avatar moves onto a bank square, the game detects their co-location through collision detection, which is a check for overlap between the bounding boxes of the two objects. The collision detection occurs within the doSomething() function of the player object, which is called once per game tick.

If the collision detection determines that the player avatar is overlapping with a bank square, the player's doSomething() function will call the landingOnSquare() function of the bank square object. This function adds the current balance of the bank to the player's coin count and resets the bank balance to 0.

After landingOnSquare() is called, the player's doSomething() function will then call the actionMoving() function of the bank square object. This function checks the player's coin count and either deducts 5 coins and adds them to the bank balance or adds the player's entire balance to the bank balance, depending on whether the player has at least 5 coins.

Finally, the appropriate sound effect is played and control returns to the player's doSomething() function, where any necessary actions are taken based on the new state of the player's coins and bank balance.

1. A list of all functionality that you failed to finish as well as known bugs in your classes, e.g. “I didn’t implement the Vortex class.” or “My Bowser doesn’t work correctly yet so it behaves like a Boo right now.”

N/A. I believe I have implemented a fully functioning game.

1. A list of assumptions you made; e.g., “It was not specified what to do in situation X, so this is what I decided to do.”

N/A. The spec was clear to me.