Snake has been a very instructive application for me to write – it was not only my first major XNA project, but also the first major project that I've completed in C#. I've found that the language shares many syntactical similarities with C++ and Java, so I am not having a hard time learning the language.

I did, on the other hand, have quite a bit of difficulty creating the snake's body. After a few long debugging sessions, I found this to be the fault of my non-rigorous approach to creating the snake's data structure. I had built an array of Vector2D's for the snake to keep track of its positions, and I was overwriting that data each time I tried to move the snake forward each update. What this resulted in was a very bizarre, short snake that had a head and that was about it – all of the snake elements inhabited the head position! I finally resolved this issue by creating a short, self-contained, correct example using only the initial snake iterates. After setting out to do this, I quickly defined an exact method for moving the 3-parts-long snake (using a temporary storage location so that I wouldn't overwrite the data from the previous movement), and was then able to extend that to an n-parts-long snake.

After the snake was working, everything else more or less fell into place. The snake game is complete, but does not allow you to restart without restarting the application. I was easily able to add some sound effects and even additional fruits through the use of a Fruit class.

Something I learned towards the end of the coding session was about the ability to create accessors for variables where they are created. This appears to be unique to C# and I hope that I can use it more in future assignments – I had been clumsily accessing data through the spriteManager object before this. I came across this while reading about implementing

sound in the “Learning XNA 4.0” textbook, which has been very instructive and helpful throughout the time I have spent working on Snake. One such example involves an animated “SNAKE” logo for the splash screen that I unfortunately did not get to implement, because I followed the book’s early example on using Game States. In this example, you can only use GameComponents in a single state. Later in Chapter 15 the book offers a solution for this, but it fell by the wayside as I prioritized other functionality to implement instead. The spritesheet for this has been left as an asset in the “Images” folder for demonstration purposes.