* ***Author/Authors:*** Michael Moorer
* ***Motivation***: My motivation for this project was from the Lab 2 assignment where we used the map function using stacks and queues most of my ideas didn’t come from. I didn’t use much of the code from that assignment but I used the ideas of stacks and queues to traverse through my list. I wanted to do an assignment that I felt prod of and the pac man game turned into a success. There was also motivation from my friends who wanted my code to consist of a lot of features which I succeeded at.
* ***Purpose***: My program is quite simple, given the Pac man map it displays coins on the screen for the user to capture. There is a special given you find the random coin and win the game. The purpose it to show the uses of stacks in python and be interactive
* ***Audience***: My intended audience is anyone who is willing to play my game. It’s fun and interactive for users of all age. It also has some humor in the pac man shape being my face
* ***Instructions:*** The game prints the instructions on the terminal. (WASD) are the keys used to maneuver the turtle and the terminal asks you to find a random point on the map and once you find that, you win the game. Once You win the game, the turtle will do a special move that is fun and entertaining.

***Design***:

|  |  |
| --- | --- |
| **Class name: Indexing** | **Index** |
| **Class Data:** | **Class Collaborations (other classes):** |
| * Stack list #1 * Stack list #2 * Stack list #3 * Integer #1 * Intger #2 * Turtle | * self |
| **Class Methods:** | **Class Collaborations (other classes):** |
| * \_init\_() # initializes stacks * Check\_dots() #places values on mp * shift() # shifts Y values * reset\_shift()# moves X bck to intit position | * self |

|  |  |
| --- | --- |
| **Class name: Class** | **class** |
| **Class Data:** | **Class Collaborations (other classes):** |
| * Stack #1 * Stack #2 * Turtle * Position #1 * Position #2 | * self |
| **Class Methods:** | **Class Collaborations (other classes):** |
| * Check\_dots() #finds position turtle cant move | * self |

* ***Enhancement***: I created a realtime map the turtle can traverse thru. I made a new function the stack list that gets some random value from the list. I also use the pop from a stack list to define boards.
* ***Functionality***: Uses my face a python turtle and moves through the map. After finding special point my face spins and grows bigger.
* ***Files***:
  + ***Class.py***
  + ***Grid4.gif***
  + ***Indexing.py***
  + ***Me.gif***
  + ***Me2.gif***
  + ***Me3.gif***
  + ***Stack()***
  + ***Moorerm-FinalProject.py***
* ***Utilized Data Structures:***  I used stack list learned during this course. The stack list was used for popping positions for the turtle when going into a non-voidable position. I could have used a Queue if necessary and could have coded the pop function slightly differently.
* ***Big O Analysis:*** I used the /stack list function as my Big O analysis. I created a definition in my class that picked a random variable given a stacked list and Big O computation is 0(n), Big O analysis is the least computation and the stck is giveni in some random order.
* ***Resources***: I only used the code learned from the course and the python library.
* ***Challenges***: Getting the values to compare correctly and finding every position on the coordinate system manually. I also had trouble keeping up with ll the values and lists.
* ***Testing***:
  + ***pathX- testing stack***
  + ***pathY-testing stack***
* ***Errors***:
  + ***The borders are not coded so it will exceed the black borders but the random point will never be ther***e.
* ***Measures and Assessment***: I believe I meet the requirements and more by making the game fun and inviting. It was colorful and creative. My coordinate system matched the X and Y coordinate systems.
* ***Summary***: I didn’t do much designing beforehand except for plotting out the plots needed for my program x and y coordinates to work properly. I however used a lot of unit testing in order to test my points and the codes functionality. My plan meant to directly mimic a pac man game with the ghosts traversing through the list. But it moved into something greater, my own game because I was short for time.
* ***I spent several hours coding, testing, and designing the map***. The hardest part of the program was coding the position on the map where there were block points. I wanted to design it dynamically where it wouldn’t have required a lot of lines of code. I did my best in doing so but some of the code required me to simply enter the values in manually since the program couldn’t directly read the image. ***Comments***: I really enjoyed doing this project even though It took me hours to code. This was the Final Project and I wanted to put my knowledge I have learned in 236 to the test. It was a final project that I can definitely say that I am proud that I have done. I am really satisfied with the final result of the project. Im glad I got the opportunity to complete this project