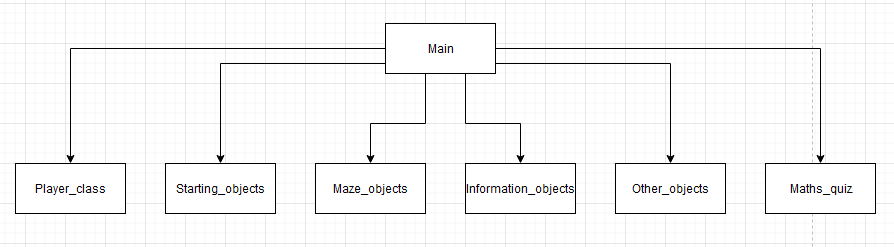
Overview

This section will show a few diagrams and sketches showing the structure of my program.

The program’s core objectives will be fulfilled using the following features:

* Start Menu (where user can select a few options including controls, instructions, quit game and also play game)
* Main game (where Score, Moves Left, Lives Left and Time Playing displayed)
* Maths Questions (where question from chosen file displayed when Moves Left are 0)
* Save game and Load game (pulls up a new screen)
* Undo moves (only if the user crashes into a wall and gets reset)
* Win screen and Lose screen (shown depending if user loses all lives or reaches end of maze)
* Hints system (where user will be shown correct path in exchange for points deducted off the final score)

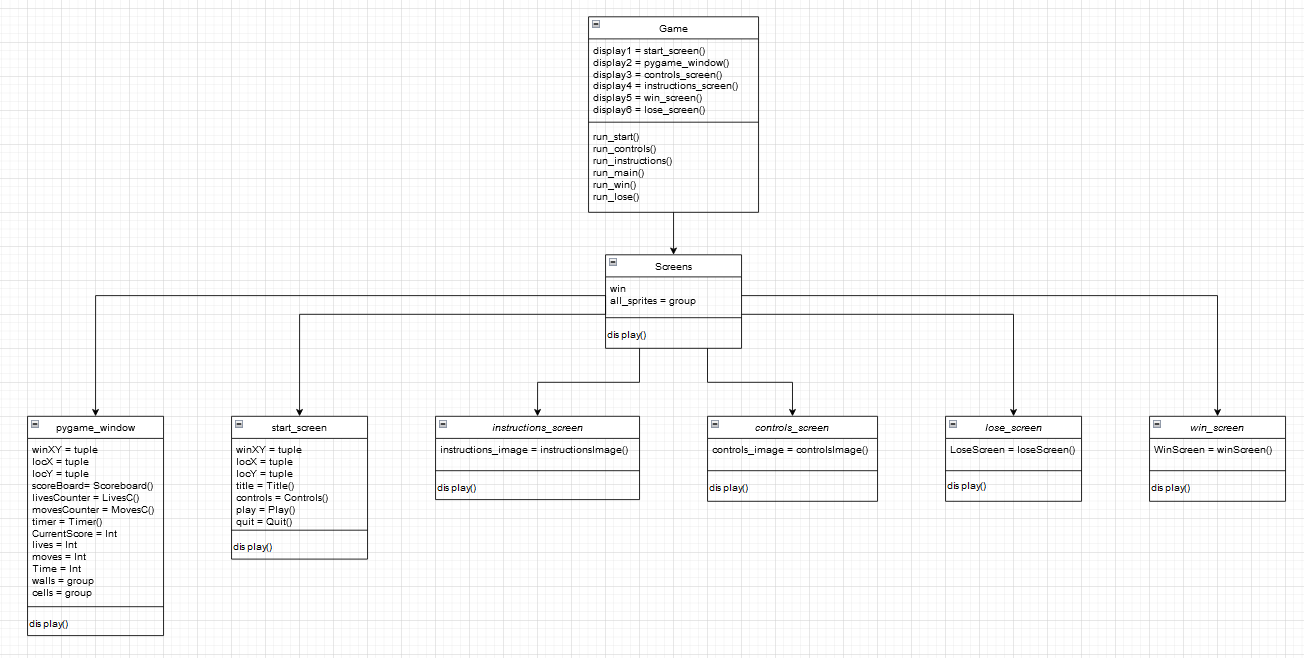
Below is a diagram of the approximate structure of the programs and subprograms:



Next, shown below is an approximate class diagram of the Main program with all the attributes and methods:

A more clear version of this image can be found in my google drive

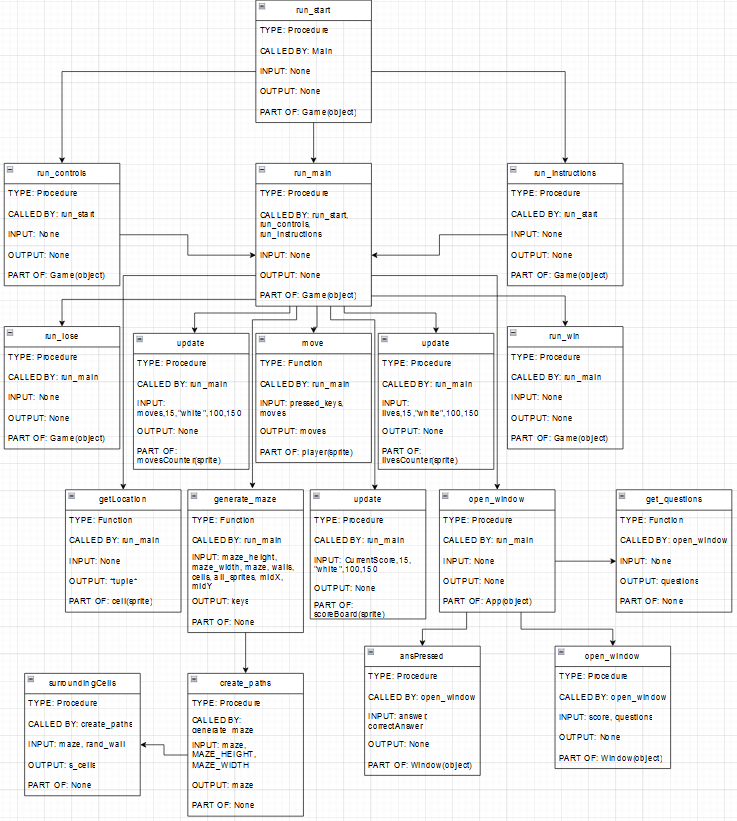
<https://drive.google.com/file/d/1tyy97dUy4hphDkmBmRAynzaBFp51_fpN/view>



Next, shown below is diagram of the subroutine/class method calls in the program:

A more clear version of this image can be found in my google drive

<https://drive.google.com/file/d/1FalLM1YocDPUfGDF47Tkg9wT-6f4ApHW/view?usp=sharing>



Next shown below is a table of all the variables in the program and information about them

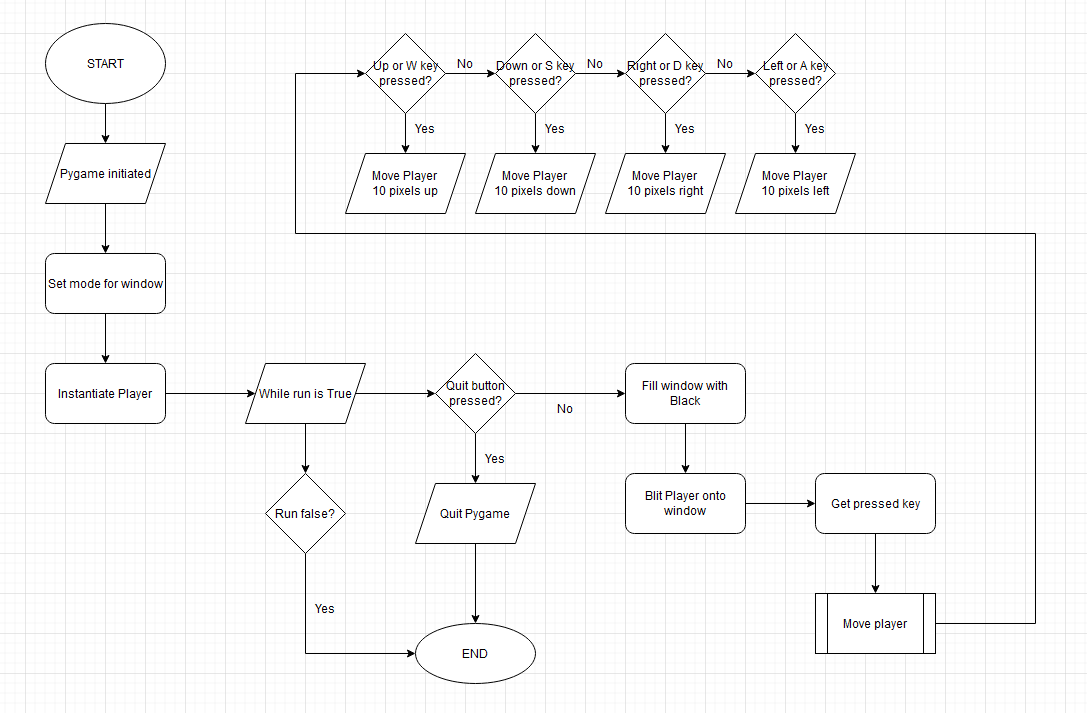
| **Variable** | **Type** | **Data Type** | **Class/Method** | **Program** |
| --- | --- | --- | --- | --- |
| savedMaze | Attribute | Array | GameSave (Constructor) | GameSave\_class |
| CurrentScore | Attribute | Integer | GameSave (Constructor) | GameSave\_class |
| lives | Attribute | Integer | GameSave (Constructor) | GameSave\_class |
| moves | Attribute | Integer | GameSave (Constructor) | GameSave\_class |
| Time | Attribute | Integer | GameSave (Constructor) | GameSave\_class |
| X | Attribute | Float | GameSave (calcPlayerPos) | GameSave\_class |
| Y | Attribute | Float | GameSave (calcPlayerPos) | GameSave\_class |
| coords | Attribute | Tuple (Integers) | GameSave (calcPlayerPos) | GameSave\_class |
| fileLines | Attribute | Array | GameSave (loadMaze) | GameSave\_class |
| maze | Attribute | Array | GameSave (loadMaze) | GameSave\_class |
| playerPos | Attribute | Tuple (Integers) | GameSave (loadMaze) | GameSave\_class |
| scores | Attribute | Array | HighScores (Constructor) | HighScores\_class |
| nameDict | Attribute | Dictionary | HighScores (Constructor) | HighScores\_class |
| line | Attribute | Array | HighScores (get\_scores) | HighScores\_class |
| m | Attribute | Integer | HighScores (sort\_scores) | HighScores\_class |
| scores1 | Attribute | Array | HighScores (sort\_scores) | HighScores\_class |
| scores2 | Attribute | Array | HighScores (sort\_scores) | HighScores\_class |
| i | Attribute | Integer | HighScores (sort\_scores) | HighScores\_class |
| j | Attribute | Integer | HighScores (sort\_scores) | HighScores\_class |
| k | Attribute | Integer | HighScores (sort\_scores) | HighScores\_class |
| sorted\_names | Attribute | Array | HighScores (sortNames) | HighScores\_class |
| last | Attribute | Integer | HighScores (check\_score) | HighScores\_class |
| names | Attribute | Array | HighScores (check\_score) | HighScores\_class |
| lastName | Attribute | String | HighScores (check\_score) | HighScores\_class |
| name | Attribute | String | HighScores (saveScore) | HighScores\_class |
| score | Attribute | Integer | HighScores (saveScore) | HighScores\_class |
| text | Attribute | String | HighScores (saveScore) | HighScores\_class |
| score | Variable | Integer | N/A | HighScores\_test |
| high\_scores | Variable | Class | N/A | HighScores\_test |
| fileIn | Variable | File | N/A | HighScores\_test |
| scores | Variable | Array | N/A | HighScores\_test |
| intScores | Variable | Array | N/A | HighScores\_test |
| name | Variable | String | N/A | HighScores\_test |
| sortedNames | Variable | Array | N/A | HighScores\_test |
| fileOut | Variable | File | N/A | HighScores\_test |
| CurrentScore | Attribute | Integer | Scoreboard (Constructor) | Information\_objects |
| score | Attribute | String | Scoreboard (Constructor) | Information\_objects |
| font | Attribute | Font | Scoreboard (Constructor) | Information\_objects |
| textSurf | Attribute | Text Surface | Scoreboard (Constructor) | Information\_objects |
| surf | Attribute | Sprite Surface | Scoreboard (Constructor) | Information\_objects |
| rect | Attribute | Sprite Rectangle | Scoreboard (Constructor) | Information\_objects |
| W | Attribute | Integer | Scoreboard (Constructor) | Information\_objects |
| H | Attribute | Integer | Scoreboard (Constructor) | Information\_objects |
| score | Attribute | String | Scoreboard (update) | Information\_objects |
| font | Attribute | Font | Scoreboard (update) | Information\_objects |
| textSurf | Attribute | Text Surface | Scoreboard (update) | Information\_objects |
| surf | Attribute | Sprite Surface | Scoreboard (update) | Information\_objects |
| W | Attribute | Integer | Scoreboard (update) | Information\_objects |
| H | Attribute | Integer | Scoreboard (update) | Information\_objects |
| Lives | Attribute | Integer | LivesC (Constructor) | Information\_objects |
| lives | Attribute | String | LivesC (Constructor) | Information\_objects |
| font | Attribute | Font | LivesC (Constructor) | Information\_objects |
| textSurf | Attribute | Text Surface | LivesC (Constructor) | Information\_objects |
| surf | Attribute | Sprite Surface | LivesC (Constructor) | Information\_objects |
| rect | Attribute | Sprite Rectangle | LivesC (Constructor) | Information\_objects |
| W | Attribute | Integer | LivesC (Constructor) | Information\_objects |
| H | Attribute | Integer | LivesC (Constructor) | Information\_objects |
| Lives | Attribute | Integer | LivesC (update) | Information\_objects |
| lives | Attribute | String | LivesC (update) | Information\_objects |
| font | Attribute | Font | LivesC (update) | Information\_objects |
| textSurf | Attribute | Text Surface | LivesC (update) | Information\_objects |
| surf | Attribute | Sprite Surface | LivesC (update) | Information\_objects |
| W | Attribute | Integer | LivesC (update) | Information\_objects |
| H | Attribute | Integer | LivesC (update) | Information\_objects |
| lives | Attribute | String | LivesC (set) | Information\_objects |
| font | Attribute | Font | LivesC (set) | Information\_objects |
| textSurf | Attribute | Text Surface | LivesC (set) | Information\_objects |
| surf | Attribute | Sprite Surface | LivesC (set) | Information\_objects |
| W | Attribute | Integer | LivesC (set) | Information\_objects |
| H | Attribute | Integer | LivesC (set) | Information\_objects |
| Moves | Attribute | Integer | MovesC (Constructor) | Information\_objects |
| moves | Attribute | String | MovesC (Constructor) | Information\_objects |
| font | Attribute | Font | MovesC (Constructor) | Information\_objects |
| textSurf | Attribute | Text Surface | MovesC (Constructor) | Information\_objects |
| surf | Attribute | Sprite Surface | MovesC (Constructor) | Information\_objects |
| rect | Attribute | Sprite Rectangle | MovesC (Constructor) | Information\_objects |
| W | Attribute | Integer | MovesC (Constructor) | Information\_objects |
| H | Attribute | Integer | MovesC (Constructor) | Information\_objects |
| moves | Attribute | String | MovesC (update) | Information\_objects |
| font | Attribute | Font | MovesC (update) | Information\_objects |
| textSurf | Attribute | Text Surface | MovesC (update) | Information\_objects |
| surf | Attribute | Sprite Surface | MovesC (update) | Information\_objects |
| W | Attribute | Integer | MovesC (update) | Information\_objects |
| H | Attribute | Integer | MovesC (update) | Information\_objects |
| Time | Attribute | Integer | Timer (Constructor) | Information\_objects |
| time | Attribute | String | Timer (Constructor) | Information\_objects |
| font | Attribute | Font | Timer (Constructor) | Information\_objects |
| textSurf | Attribute | Text Surface | Timer (Constructor) | Information\_objects |
| surf | Attribute | Sprite Surface | Timer (Constructor) | Information\_objects |
| rect | Attribute | Sprite Rectangle | Timer (Constructor) | Information\_objects |
| W | Attribute | Integer | Timer (Constructor) | Information\_objects |
| H | Attribute | Integer | Timer (Constructor) | Information\_objects |
| moves | Attribute | String | Timer (update) | Information\_objects |
| font | Attribute | Font | Timer (update) | Information\_objects |
| textSurf | Attribute | Text Surface | Timer (update) | Information\_objects |
| surf | Attribute | Sprite Surface | Timer (update) | Information\_objects |
| W | Attribute | Integer | Timer (update) | Information\_objects |
| H | Attribute | Integer | Timer (update) | Information\_objects |
| surf | Attribute | Sprite Surface | SaveButton (Constructor) | Information\_objects |
| rect | Attribute | Sprite Rectangle | SaveButton (Constructor) | Information\_objects |
| image | Attribute | Sprite image | SaveButton  (Constructor) | Information\_objects |
| surf | Attribute | Sprite Surface | LoadButton (Constructor) | Information\_objects |
| rect | Attribute | Sprite Rectangle | LoadButton (Constructor) | Information\_objects |
| image | Attribute | Sprite image | LoadButton  (Constructor) | Information\_objects |
| surf | Attribute | Sprite Surface | UndoButton (Constructor) | Information\_objects |
| rect | Attribute | Sprite Rectangle | UndoButton (Constructor) | Information\_objects |
| image | Attribute | Sprite image | UndoButton  (Constructor) | Information\_objects |
| questions | Variable | Array | get\_questions | Maths\_quiz |
| file | Variable | File | get\_questions | Maths\_quiz |
| newLine | Variable | Array | get\_questions | Maths\_quiz |
| label1 | Attribute | Tkinter Label | Results (Constructor) | Maths\_quiz |
| thisText | Attribute | String | Results (Constructor) | Maths\_quiz |
| label2 | Attribute | Tkinter Label | Results (Constructor) | Maths\_quiz |
| question | Attribute | Array | Window (Constructor) | Maths\_quiz |
| thisQuestion | Attribute | String | Window (Constructor) | Maths\_quiz |
| answer1 | Attribute | String | Window (Constructor) | Maths\_quiz |
| answer2 | Attribute | String | Window (Constructor) | Maths\_quiz |
| answer3 | Attribute | String | Window (Constructor) | Maths\_quiz |
| answer4 | Attribute | String | Window (Constructor) | Maths\_quiz |
| correctAnswer | Attribute | Array | Window (Constructor) | Maths\_quiz |
| Score | Attribute | String Var | Window (Constructor) | Maths\_quiz |
| movesLeft | Attribute | Integer | Window (Constructor) | Maths\_quiz |
| currentScore | Attribute | Integer | Window (Constructor) | Maths\_quiz |
| label1 | Attribute | Tkinter Label | Window (Constructor) | Maths\_quiz |
| number1 | Attribute | Tkinter Button | Window (Constructor) | Maths\_quiz |
| number2 | Attribute | Tkinter Button | Window (Constructor) | Maths\_quiz |
| number3 | Attribute | Tkinter Button | Window (Constructor) | Maths\_quiz |
| number4 | Attribute | Tkinter Button | Window (Constructor) | Maths\_quiz |
| scoreBox | Attribute | Tkinter Entry | Window (Constructor) | Maths\_quiz |
| label2 | Attribute | Tkinter Label | Window (Constructor) | Maths\_quiz |
| window | Attribute | Class | Window (open\_window) | Maths\_quiz |
| results | Attribute | Class | Window (open\_window) | Maths\_quiz |
| label1 | Attribute | Tkinter Label | App (Constructor) | Maths\_quiz |
| calls | Attribute | Integer | App (open\_window) | Maths\_quiz |
| questions | Attribute | Array | App (open\_window) | Maths\_quiz |
| window | Attribute | Class | App (open\_window) | Maths\_quiz |
| score | Variable | Integer | N/A | Maths\_quiz |
| calls | Variable | Integer | N/A | Maths\_quiz |
| app | Variable | Class | N/A | Maths\_quiz |
| surf | Attribute | Sprite Surface | Wall (Constructor) | Maze\_objects |
| rect | Attribute | Sprite Rectangle | Wall (Constructor) | Maze\_objects |
| surf | Attribute | Sprite Surface | Cell (Constructor) | Maze\_objects |
| location | Attribute | Tuple (Integers) | Cell (Constructor) | Maze\_objects |
| rect | Attribute | Sprite Rectangle | Cell (Constructor) | Maze\_objects |
| surf | Attribute | Sprite Surface | Empty (Constructor) | Maze\_objects |
| rect | Attribute | Sprite Rectangle | Empty (Constructor) | Maze\_objects |
| surf | Attribute | Sprite Surface | Key (Constructor) | Maze\_objects |
| image | Attribute | Sprite image | Key  (Constructor) | Maze\_objects |
| rect | Attribute | Sprite Rectangle | Key (Constructor) | Maze\_objects |
| surf | Attribute | Sprite Surface | controlsImage (Constructor) | Other\_objects |
| image | Attribute | Sprite image | controlsImage  (Constructor) | Other\_objects |
| rect | Attribute | Sprite Rectangle | controlsImage (Constructor) | Other\_objects |
| surf | Attribute | Sprite Surface | instructionsImage (Constructor) | Other\_objects |
| image | Attribute | Sprite image | instructionsImage  (Constructor) | Other\_objects |
| rect | Attribute | Sprite Rectangle | instructionsImage (Constructor) | Other\_objects |
| surf | Attribute | Sprite Surface | winScreen (Constructor) | Other\_objects |
| image | Attribute | Sprite image | winScreen  (Constructor) | Other\_objects |
| rect | Attribute | Sprite Rectangle | winScreen (Constructor) | Other\_objects |
| surf | Attribute | Sprite Surface | loseScreen (Constructor) | Other\_objects |
| image | Attribute | Sprite image | loseScreen  (Constructor) | Other\_objects |
| rect | Attribute | Sprite Rectangle | loseScreen (Constructor) | Other\_objects |
| surf | Attribute | Sprite Surface | SS\_image (Constructor) | Other\_objects |
| image | Attribute | Sprite image | SS\_image  (Constructor) | Other\_objects |
| rect | Attribute | Sprite Rectangle | SS\_image (Constructor) | Other\_objects |
| surf | Attribute | Sprite Surface | LS\_image (Constructor) | Other\_objects |
| image | Attribute | Sprite image | LS\_image  (Constructor) | Other\_objects |
| rect | Attribute | Sprite Rectangle | LS\_image (Constructor) | Other\_objects |
| surf | Attribute | Sprite Surface | Player (Constructor) | Player\_class |
| location | Attribute | Tuple (Integers) | Player (Constructor) | Player\_class |
| rect | Attribute | Sprite Rectangle | Player (Constructor) | Player\_class |
| moves | Attribute | Integer | Player (move) | Player\_class |
| movesLeft | Attribute | Integer | Player (move) | Player\_class |
| font | Attribute | Font | Text1 (Constructor) | QS\_objects |
| textSurf | Attribute | Text Surface | Text1 (Constructor) | QS\_objects |
| surf | Attribute | Sprite Surface | Text1 (Constructor) | QS\_objects |
| rect | Attribute | Sprite Rectangle | Text1 (Constructor) | QS\_objects |
| W | Attribute | Integer | Text1 (Constructor) | QS\_objects |
| H | Attribute | Integer | Text1 (Constructor) | QS\_objects |
| font | Attribute | Font | Button1 (Constructor) | QS\_objects |
| textSurf | Attribute | Text Surface | Button1 (Constructor) | QS\_objects |
| surf | Attribute | Sprite Surface | Button1 (Constructor) | QS\_objects |
| rect | Attribute | Sprite Rectangle | Button1 (Constructor) | QS\_objects |
| W | Attribute | Integer | Button1 (Constructor) | QS\_objects |
| H | Attribute | Integer | Button1 (Constructor) | QS\_objects |
| font | Attribute | Font | Button2  (Constructor) | QS\_objects |
| textSurf | Attribute | Text Surface | Button2 (Constructor) | QS\_objects |
| surf | Attribute | Sprite Surface | Button2 (Constructor) | QS\_objects |
| rect | Attribute | Sprite Rectangle | Button2 (Constructor) | QS\_objects |
| W | Attribute | Integer | Button2 (Constructor) | QS\_objects |
| H | Attribute | Integer | Button2 (Constructor) | QS\_objects |
| font | Attribute | Font | Button3 (Constructor) | QS\_objects |
| textSurf | Attribute | Text Surface | Button3 (Constructor) | QS\_objects |
| surf | Attribute | Sprite Surface | Button3 (Constructor) | QS\_objects |
| rect | Attribute | Sprite Rectangle | Button3 (Constructor) | QS\_objects |
| W | Attribute | Integer | Button3 (Constructor) | QS\_objects |
| H | Attribute | Integer | Button3 (Constructor) | QS\_objects |
| font | Attribute | Font | Button4 (Constructor) | QS\_objects |
| textSurf | Attribute | Text Surface | Button4 (Constructor) | QS\_objects |
| surf | Attribute | Sprite Surface | Button4 (Constructor) | QS\_objects |
| rect | Attribute | Sprite Rectangle | Button4 (Constructor) | QS\_objects |
| W | Attribute | Integer | Button4 (Constructor) | QS\_objects |
| H | Attribute | Integer | Button4 (Constructor) | QS\_objects |
| font | Attribute | Font | Button5 (Constructor) | QS\_objects |
| textSurf | Attribute | Text Surface | Button5 (Constructor) | QS\_objects |
| surf | Attribute | Sprite Surface | Button5 (Constructor) | QS\_objects |
| rect | Attribute | Sprite Rectangle | Button5 (Constructor) | QS\_objects |
| W | Attribute | Integer | Button5 (Constructor) | QS\_objects |
| H | Attribute | Integer | Button5 (Constructor) | QS\_objects |
| font | Attribute | Font | Title  (Constructor) | Starting\_objects |
| textSurf | Attribute | Text Surface | Title (Constructor) | Starting\_objects |
| surf | Attribute | Sprite Surface | Title (Constructor) | Starting\_objects |
| rect | Attribute | Sprite Rectangle | Title (Constructor) | Starting\_objects |
| W | Attribute | Integer | Title (Constructor) | Starting\_objects |
| H | Attribute | Integer | Title (Constructor) | Starting\_objects |
| font | Attribute | Font | Controls  (Constructor) | Starting\_objects |
| textSurf | Attribute | Text Surface | Controls (Constructor) | Starting\_objects |
| surf | Attribute | Sprite Surface | Controls (Constructor) | Starting\_objects |
| rect | Attribute | Sprite Rectangle | Controls (Constructor) | Starting\_objects |
| W | Attribute | Integer | Controls (Constructor) | Starting\_objects |
| H | Attribute | Integer | Controls (Constructor) | Starting\_objects |
| font | Attribute | Font | Play  (Constructor) | Starting\_objects |
| textSurf | Attribute | Text Surface | Play (Constructor) | Starting\_objects |
| surf | Attribute | Sprite Surface | Play (Constructor) | Starting\_objects |
| rect | Attribute | Sprite Rectangle | Play (Constructor) | Starting\_objects |
| W | Attribute | Integer | Play (Constructor) | Starting\_objects |
| H | Attribute | Integer | Play (Constructor) | Starting\_objects |
| font | Attribute | Font | Instructions  (Constructor) | Starting\_objects |
| textSurf | Attribute | Text Surface | Instructions (Constructor) | Starting\_objects |
| surf | Attribute | Sprite Surface | Instructions (Constructor) | Starting\_objects |
| rect | Attribute | Sprite Rectangle | Instructions (Constructor) | Starting\_objects |
| W | Attribute | Integer | Instructions (Constructor) | Starting\_objects |
| H | Attribute | Integer | Instructions (Constructor) | Starting\_objects |
| font | Attribute | Font | Quit  (Constructor) | Starting\_objects |
| textSurf | Attribute | Text Surface | Quit (Constructor) | Starting\_objects |
| surf | Attribute | Sprite Surface | Quit (Constructor) | Starting\_objects |
| rect | Attribute | Sprite Rectangle | Quit (Constructor) | Starting\_objects |
| W | Attribute | Integer | Quit (Constructor) | Starting\_objects |
| H | Attribute | Integer | Quit (Constructor) | Starting\_objects |
| MAZE\_HEIGHT | Constant | Integer | N/A | Main |
| MAZE\_WIDTH | Constant | Integer | N/A | Main |
| MAZE\_CONSTANT | Constant | Integer | N/A | Main |
| maze | Variable | Array | N/A | Main |
| win | Attribute | Pygame Display | screens (Constructor) | Main |
| winXY | Attribute | Integer | screens (Constructor) | Main |
| all\_sprites | Attribute | Sprite group | screens (Constructor) | Main |
| questions | Variable | Array | get\_questions | Main |
| newLine | Variable | Array | get\_questions | Main |
| s\_cells | Variable | Integer | surroundingCells | Main |
| starting\_height | Variable | Integer | surroundingCells | Main |
| starting\_width | Variable | Integer | surroundingCells | Main |
| wallCoords | Variable | Array | surroundingCells | Main |
| rand\_wall | Variable | Tuple (Integers) | surroundingCells | Main |
| visited | Variable | Array | generate\_maze | Main |
| visitedRow | Variable | Array | generate\_maze | Main |
| mazeRow | Variable | Array | generate\_maze | Main |
| X | Variable | Integer | generate\_maze | Main |
| Y | Variable | Integer | generate\_maze | Main |
| wall | Variable | Pygame Sprite | generate\_maze | Main |
| cell | Variable | Pygame Sprite | generate\_maze | Main |
| empty | Variable | Pygame Sprite | generate\_maze | Main |
| keys | Variable | Sprite group | generate\_maze | Main |
| key | Variable | Pygame Sprite | generate\_maze | Main |
| question | Attribute | Array | QuestionWindow (Constructor) | Main |
| thisQuestion | Attribute | String | QuestionWindow (Constructor) | Main |
| answer1 | Attribute | String | QuestionWindow (Constructor) | Main |
| answer2 | Attribute | String | QuestionWindow (Constructor) | Main |
| answer3 | Attribute | String | QuestionWindow (Constructor) | Main |
| answer4 | Attribute | String | QuestionWindow (Constructor) | Main |
| correctAnswer | Attribute | Array | QuestionWindow (Constructor) | Main |
| Score | Attribute | String Var | QuestionWindow (Constructor) | Main |
| movesLeft | Attribute | Integer | QuestionWindow (Constructor) | Main |
| score | Attribute | Integer | QuestionWindow (Constructor) | Main |
| status | Attribute | String | QuestionWindow (Constructor) | Main |
| label1 | Attribute | Tkinter Label | QuestionWindow (Constructor) | Main |
| number1 | Attribute | Tkinter Button | QuestionWindow (Constructor) | Main |
| number2 | Attribute | Tkinter Button | QuestionWindow (Constructor) | Main |
| number3 | Attribute | Tkinter Button | QuestionWindow (Constructor) | Main |
| number4 | Attribute | Tkinter Button | QuestionWindow (Constructor) | Main |
| scoreBox | Attribute | Tkinter Entry | QuestionWindow (Constructor) | Main |
| label2 | Attribute | Tkinter Label | QuestionWindow (Constructor) | Main |
| currentScore | Attribute | Integer | QuestionWindow (return\_info) | Main |
| winX | Attribute | Integer | QuestionSelect\_screen  (Constructor) | Main |
| winY | Attribute | Integer | QuestionSelect\_screen  (Constructor) | Main |
| midX | Attribute | Integer | QuestionSelect\_screen  (Constructor) | Main |
| midY | Attribute | Integer | QuestionSelect\_screen  (Constructor) | Main |
| locX1 | Attribute | Integer | QuestionSelect\_screen  (Constructor) | Main |
| locY1 | Attribute | Integer | QuestionSelect\_screen  (Constructor) | Main |
| locY2 | Attribute | Integer | QuestionSelect\_screen  (Constructor) | Main |
| locY3 | Attribute | Integer | QuestionSelect\_screen  (Constructor) | Main |
| locY4 | Attribute | Integer | QuestionSelect\_screen  (Constructor) | Main |
| locY5 | Attribute | Integer | QuestionSelect\_screen  (Constructor) | Main |
| locY6 | Attribute | Integer | QuestionSelect\_screen  (Constructor) | Main |
| SyQD | Attribute | Class | QuestionSelect\_screen  (Constructor) | Main |
| GCSE | Attribute | Integer | QuestionSelect\_screen  (Constructor) | Main |
| Easy | Attribute | Integer | QuestionSelect\_screen  (Constructor) | Main |
| Medium | Attribute | Integer | QuestionSelect\_screen  (Constructor) | Main |
| Hard | Attribute | Integer | QuestionSelect\_screen  (Constructor) | Main |
| NS | Attribute | Integer | QuestionSelect\_screen  (Constructor) | Main |
| loc1 | Attribute | Integer | QuestionSelect\_screen  (display) | Main |
| loc2 | Attribute | Integer | QuestionSelect\_screen  (display) | Main |
| loc3 | Attribute | Integer | QuestionSelect\_screen  (display) | Main |
| loc4 | Attribute | Integer | QuestionSelect\_screen  (display) | Main |
| loc5 | Attribute | Integer | QuestionSelect\_screen  (display) | Main |
| loc6 | Attribute | Integer | QuestionSelect\_screen  (display) | Main |
| loc7 | Attribute | Integer | QuestionSelect\_screen  (display) | Main |
| loc8 | Attribute | Integer | QuestionSelect\_screen  (display) | Main |
| loc9 | Attribute | Integer | QuestionSelect\_screen  (display) | Main |
| loc10 | Attribute | Integer | QuestionSelect\_screen  (display) | Main |
| loc11 | Attribute | Integer | QuestionSelect\_screen  (display) | Main |
| loc12 | Attribute | Integer | QuestionSelect\_screen  (display) | Main |
| run | Attribute | Boolean | QuestionSelect\_screen  (display) | Main |
| mouse | Attribute | Tuple (Integers) | QuestionSelect\_screen  (display) | Main |
| qs\_diff | Attribute | String | QuestionSelect\_screen  (display) | Main |
| high\_scores | Attribute | Class | high\_scores \_ screen2  (Constructor) | Main |
| scoreValid | Attribute | Boolean | high\_scores \_ screen2  (Constructor) | Main |
| fileIn | Attribute | File | high\_scores \_ screen2  (Constructor) | Main |
| sortedNames | Attribute | Array | high\_scores \_ screen2  (Constructor) | Main |
| fileOut | Attribute | File | high\_scores \_ screen2  (Constructor) | Main |
| label1 | Attribute | Tkinter Label | high\_scores \_ screen2  (Constructor) | Main |
| textBox | Attribute | Tkinter Textbox | high\_scores \_ screen2  (Constructor) | Main |
| nameScore | Attribute | Tuple (Strings) | high\_scores \_ screen2  (Constructor) | Main |
| name | Attribute | String | high\_scores \_ screen2  (Constructor) | Main |
| score | Attribute | Floating Point (Real) | high\_scores \_ screen2  (Constructor) | Main |
| text | Attribute | String | high\_scores \_ screen2  (Constructor) | Main |
| text1 | Attribute | String | high\_scores \_ screen2  (displayInfo) | Main |
| text2 | Attribute | String | high\_scores \_ screen2  (displayInfo) | Main |
| text3 | Attribute | String | high\_scores \_ screen2  (displayInfo) | Main |
| text4 | Attribute | String | high\_scores \_ screen2  (displayInfo) | Main |
| text | Attribute | String | high\_scores \_ screen1  (Constructor) | Main |
| label1 | Attribute | Tkinter Label | high\_scores \_ screen1  (Constructor) | Main |
| label2 | Attribute | Tkinter Label | high\_scores \_ screen1  (Constructor) | Main |
| entry1 | Attribute | Tkinter Entry | high\_scores \_ screen1  (Constructor) | Main |
| textBox | Attribute | Tkinter Textbox | high\_scores \_ screen1  (Constructor) | Main |
| name | Attribute | String | high\_scores \_ screen1  (open\_window) | Main |
| text | Attribute | String | high\_scores \_ screen1  (open\_window) | Main |
| hss2 | Attribute | Class | high\_scores \_ screen1  (open\_window) | Main |
| SaveScreen | Attribute | Class | save\_screen  (Constructor) | Main |
| gameSave | Attribute | Class | save\_screen  (Constructor) | Main |
| file | Attribute | File | save\_screen  (save) | Main |
| savedMaze | Attribute | Array | load\_screen (Constructor) | Main |
| CurrentScore | Attribute | Integer | load\_screen (Constructor) | Main |
| lives | Attribute | Integer | load\_screen (Constructor) | Main |
| moves | Attribute | Integer | load\_screen (Constructor) | Main |
| Time | Attribute | Integer | load\_screen (Constructor) | Main |
| playerPos | Attribute | Tuple (Integers) | load\_screen (Constructor) | Main |
| midX | Attribute | Integer | load\_screen (Constructor) | Main |
| midY | Attribute | Integer | load\_screen (Constructor) | Main |
| LoadScreen | Attribute | Class | load\_screen (Constructor) | Main |
| all\_sprites | Attribute | Sprite group | load\_screen (loadMaze) | Main |
| walls | Attribute | Sprite group | load\_screen (loadMaze) | Main |
| cells | Attribute | Sprite group | load\_screen (loadMaze) | Main |
| X | Attribute | Integer | load\_screen (loadMaze) | Main |
| Y | Attribute | Integer | load\_screen (loadMaze) | Main |
| wall | Attribute | Class | load\_screen (loadMaze) | Main |
| cell | Attribute | Class | load\_screen (loadMaze) | Main |
| empty | Attribute | Class | load\_screen (loadMaze) | Main |
| new\_sprites | Attribute | Sprite group | load\_screen (loadGame) | Main |
| pixelX | Attribute | Integer | load\_screen (loadGame) | Main |
| pixelY | Attribute | Integer | load\_screen (loadGame) | Main |
| player | Attribute | Class | load\_screen (loadGame) | Main |
| locX1 | Attribute | Integer | load\_screen (loadGame) | Main |
| locX2 | Attribute | Integer | load\_screen (loadGame) | Main |
| locX3 | Attribute | Integer | load\_screen (loadGame) | Main |
| locX4 | Attribute | Integer | load\_screen (loadGame) | Main |
| locY | Attribute | Integer | load\_screen (loadGame) | Main |
| scoreBoard | Attribute | Class | load\_screen (loadGame) | Main |
| livesCounter | Attribute | Class | load\_screen (loadGame) | Main |
| movesCounter | Attribute | Class | load\_screen (loadGame) | Main |
| timer | Attribute | Class | load\_screen (loadGame) | Main |
| WinScreen | Attribute | Class | win\_screen (Constructor) | Main |
| LoseScreen | Attribute | Class | lose\_screen (Constructor) | Main |
| controls\_image | Attribute | Class | controls\_screen(Constructor) | Main |
| instructions\_image | Attribute | Class | instructions\_screen (Constructor) | Main |
| winX | Attribute | Integer | start\_screen (Constructor) | Main |
| winY | Attribute | Integer | start\_screen (Constructor) | Main |
| midX | Attribute | Integer | start\_screen (Constructor) | Main |
| midY | Attribute | Integer | start\_screen (Constructor) | Main |
| locX1 | Attribute | Integer | start\_screen (Constructor) | Main |
| locX2 | Attribute | Integer | start\_screen (Constructor) | Main |
| locX3 | Attribute | Integer | start\_screen (Constructor) | Main |
| locX4 | Attribute | Integer | start\_screen (Constructor) | Main |
| locX5 | Attribute | Integer | start\_screen (Constructor) | Main |
| locY1 | Attribute | Integer | start\_screen (Constructor) | Main |
| locY2 | Attribute | Integer | start\_screen (Constructor) | Main |
| title | Attribute | Class | start\_screen (Constructor) | Main |
| controls | Attribute | Class | start\_screen (Constructor) | Main |
| play | Attribute | Class | start\_screen (Constructor) | Main |
| instructions | Attribute | Class | start\_screen (Constructor) | Main |
| quit | Attribute | Class | start\_screen (Constructor) | Main |
| loc1 | Attribute | Integer | start\_screen (display) | Main |
| loc2 | Attribute | Integer | start\_screen (display) | Main |
| loc3 | Attribute | Integer | start\_screen (display) | Main |
| loc4 | Attribute | Integer | start\_screen (display) | Main |
| loc5 | Attribute | Integer | start\_screen (display) | Main |
| loc6 | Attribute | Integer | start\_screen (display) | Main |
| loc7 | Attribute | Integer | start\_screen (display) | Main |
| loc8 | Attribute | Integer | start\_screen (display) | Main |
| loc9 | Attribute | Integer | start\_screen (display) | Main |
| loc0 | Attribute | Integer | start\_screen (display) | Main |
| mouse | Attribute | Tuple (Integers) | start\_screen (display) | Main |
| maze\_height | Attribute | Integer | pygame\_window (Constructor) | Main |
| maze\_width | Attribute | Integer | pygame\_window (Constructor) | Main |
| maze | Attribute | Array | pygame\_window (Constructor) | Main |
| winX | Attribute | Integer | pygame\_window (Constructor) | Main |
| winY | Attribute | Integer | pygame\_window (Constructor) | Main |
| midX | Attribute | Integer | pygame\_window (Constructor) | Main |
| midY | Attribute | Integer | pygame\_window (Constructor) | Main |
| locX1 | Attribute | Integer | pygame\_window (Constructor) | Main |
| locX2 | Attribute | Integer | pygame\_window (Constructor) | Main |
| locX3 | Attribute | Integer | pygame\_window (Constructor) | Main |
| locX4 | Attribute | Integer | pygame\_window (Constructor) | Main |
| locY | Attribute | Integer | pygame\_window (Constructor) | Main |
| locY1 | Attribute | Integer | pygame\_window (Constructor) | Main |
| locY2 | Attribute | Integer | pygame\_window (Constructor) | Main |
| sWidth | Attribute | Integer | pygame\_window (Constructor) | Main |
| sHeight | Attribute | Integer | pygame\_window (Constructor) | Main |
| player | Attribute | Class | pygame\_window (Constructor) | Main |
| scoreBoard | Attribute | Class | pygame\_window (Constructor) | Main |
| livesCounter | Attribute | Class | pygame\_window (Constructor) | Main |
| movesCounter | Attribute | Class | pygame\_window (Constructor) | Main |
| timer | Attribute | Class | pygame\_window (Constructor) | Main |
| saveButton | Attribute | Class | pygame\_window (Constructor) | Main |
| loadButton | Attribute | Class | pygame\_window (Constructor) | Main |
| undoButton | Attribute | Class | pygame\_window (Constructor) | Main |
| hintButton | Attribute | Class | pygame\_window (Constructor) | Main |
| CurrentScore | Attribute | Integer | pygame\_window (Constructor) | Main |
| lives | Attribute | Integer | pygame\_window (Constructor) | Main |
| moves | Attribute | Integer | pygame\_window (Constructor) | Main |
| movesLeft | Attribute | Integer | pygame\_window (Constructor) | Main |
| Time | Attribute | Integer | pygame\_window (Constructor) | Main |
| coords | Attribute | Tuple (Integers) | pygame\_window (Constructor) | Main |
| file | Attribute | File | pygame\_window (Constructor) | Main |
| gameSave | Attribute | Class | pygame\_window (Constructor) | Main |
| movesQueue | Attribute | Array | pygame\_window (Constructor) | Main |
| walls | Attribute | Sprite Group | pygame\_window (Constructor) | Main |
| cells | Attribute | Sprite Group | pygame\_window (Constructor) | Main |
| keys | Attribute | Sprite Group | pygame\_window (display) | Main |
| visited | Attribute | Array | pygame\_window (display) | Main |
| maze | Attribute | Array | pygame\_window (display) | Main |
| loc1 | Attribute | Integer | pygame\_window (display) | Main |
| loc2 | Attribute | Integer | pygame\_window (display) | Main |
| loc3 | Attribute | Integer | pygame\_window (display) | Main |
| loc4 | Attribute | Integer | pygame\_window (display) | Main |
| loc5 | Attribute | Integer | pygame\_window (display) | Main |
| loc6 | Attribute | Integer | pygame\_window (display) | Main |
| loc7 | Attribute | Integer | pygame\_window (display) | Main |
| loc8 | Attribute | Integer | pygame\_window (display) | Main |
| loc9 | Attribute | Integer | pygame\_window (display) | Main |
| loc10 | Attribute | Integer | pygame\_window (display) | Main |
| score | Attribute | Integer | pygame\_window (display) | Main |
| question | Attribute | Array | pygame\_window (display) | Main |
| calls | Attribute | Integer | pygame\_window (display) | Main |
| crashes | Attribute | Integer | pygame\_window (display) | Main |
| devMode | Attribute | Boolean | pygame\_window (display) | Main |
| moveUndone | Attribute | Boolean | pygame\_window (display) | Main |
| hintRevealed | Attribute | Boolean | pygame\_window (display) | Main |
| SS | Attribute | Class | pygame\_window (display) | Main |
| LS | Attribute | Class | pygame\_window (display) | Main |
| pressed\_keys | Attribute | String | pygame\_window (display) | Main |
| mouse | Attribute | Tuple (Integers) | pygame\_window (display) | Main |
| hint | Attribute | Tuple (Integers) | pygame\_window (display) | Main |
| hintX | Attribute | Integer | pygame\_window (display) | Main |
| hintY | Attribute | Integer | pygame\_window (display) | Main |
| cellLoc | Attribute | Tuple (Integers) | pygame\_window (display) | Main |
| movesBonus | Attribute | Integer | pygame\_window (display) | Main |
| questionsBonus | Attribute | Integer | pygame\_window (display) | Main |
| livesBonus | Attribute | Integer | pygame\_window (display) | Main |
| window | Attribute | Class | pygame\_window (open\_question) | Main |
| question | Attribute | Array | pygame\_window (open\_question) | Main |
| lastMove | Attribute | Tuple (Integers) | pygame\_window (undoMove) | Main |
| moveSafe | Attribute | Boolean | pygame\_window (checkQueue) | Main |
| qs\_diff | Attribute | String | Game (Constructor) | Main |
| display1 | Attribute | Class | Game (Constructor) | Main |
| display2 | Attribute | Class | Game (Constructor) | Main |
| display3 | Attribute | Class | Game (Constructor) | Main |
| display4 | Attribute | Class | Game (Constructor) | Main |
| display5 | Attribute | Class | Game (Constructor) | Main |
| display6 | Attribute | Class | Game (Constructor) | Main |
| display7 | Attribute | Class | Game (Constructor) | Main |
| qs\_diff | Variable | String | N/A | Main |
| game | Variable | Class | N/A | Main |

Player class

This first thing I am going to do when I make my program is create the Player class

This will allow the player to spawn and respawn as a rectangular sprite and also move around when the user presses WASD or the arrow keys.

Below is a flowchart showing how the Player class will fit into the Main program:



***NOTE: The syntax/format for any pseudocode can be found in Appendix B0***

After creating the Player class (pseudocode can be found in Appendix B1) I am going to instantiate the player and set up a pygame loop in the main program which will just blit the player onto the pygame screen and allow the user to move the player sprite around. Below is the pseudocode for this.

IMPORT ***pygame***

from time IMPORT ***sleep***

from Player\_class IMPORT Player

***pygame***.init()

win ← ***pygame***.display.set\_mode((1900, 1000), pygame.FULLSCREEN)

***pygame***.display.set\_caption("Maths Maze Navigator")

player ← new Player("Player.png")

run ← **True**

WHILE run = **True**

FOR event IN ***pygame***.event.get()

IF event.type = ***pygame***.QUIT THEN

run ← **False**

ENDIF

ENDFOR

win.*fill*((0,0,0))

win.*blit*(player.surf, player.rect)

***pygame***.display.update()

pressed\_keys = ***pygame***.key.get\_pressed()

player.move(pressed\_keys)

ENDWHILE

IF run = **False** THEN

***sleep(2)***

***pygame***.quit()

ENDIF

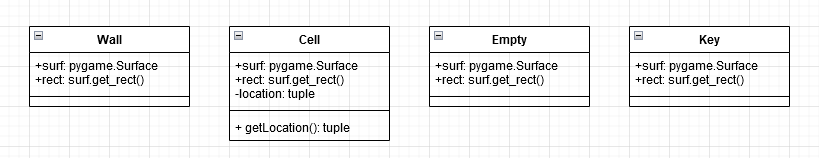
After that, I am going to create my maze objects which will include cell sprites, wall sprites, an empty sprite for the entrance and exit of the maze and also a key sprite that the user can collect to end the game.

Maze Objects

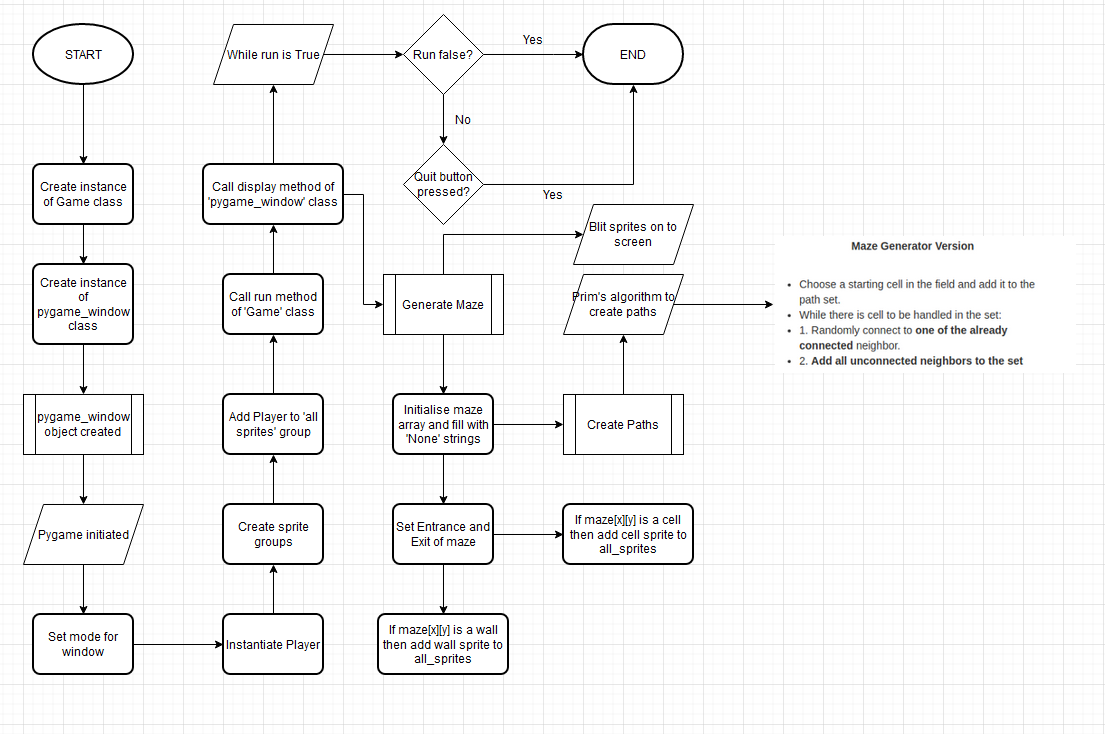
The next thing I am going to do is create the Cell, Wall and Empty and Key classes.

These will be done in a subprogram called ‘Maze\_objects’

Below is the class diagram for this subprogram:



Below is the flowchart of the updated main program after the maze objects have been added:



The pseudocode for the Maze\_objects subprogram can be found in Appendix B2

On the next page, is the Pseudocode for the Main program after it has been updated

Prim’s algorithm implementation (in create\_paths function) was found on github

<https://github.com/OrWestSide/python-scripts/blob/master/maze.py>

IMPORT ***pygame***

FROM time IMPORT ***sleep***

FROM random IMPORT ***randint***,***random***

FROM Player\_class IMPORT Player

FROM Maze\_objects IMPORT Wall, Cell, Empty

MAZE\_HEIGHT ← 10

MAZE\_WIDTH ← 10

maze ← []

FUNCTION surroundingCells(maze, rand\_wall)

s\_cells ← 0

IF (maze[rand\_wall[0]-1][rand\_wall[1]] = **True**) THEN

s\_cells ← s\_cells + 1

ENDIF

IF (maze[rand\_wall[0]+1][rand\_wall[1]] = **True**) THEN

s\_cells ← s\_cells + 1

ENDIF

IF (maze[rand\_wall[0]][rand\_wall[1]-1] = **True**) THEN

s\_cells ← s\_cells + 1

ENDIF

IF (maze[rand\_wall[0]][rand\_wall[1]+1] = **True**) THEN

s\_cells ← s\_cells + 1

ENDIF

RETURN s\_cells

ENDFUNCTION

FUNCTION create\_paths(maze, MAZE\_HEIGHT, MAZE\_WIDTH)

starting\_height ← *Int*(***random()***\*MAZE\_HEIGHT)

starting\_width ← *Int*(***random()***\*MAZE\_WIDTH)

IF (starting\_height = 0) THEN

starting\_height ← starting\_height + 1

ENDIF

IF (starting\_height = MAZE\_HEIGHT-1) THEN

starting\_height ← starting\_height - 1

ENDIF

IF (starting\_width = 0) THEN

starting\_width ← starting\_width + 1

ENDIF

IF (starting\_width = MAZE\_WIDTH-1) THEN

starting\_width ← starting\_width - 1

ENDIF

maze[starting\_height][starting\_width] ← **True**

wallCoords ← []

wallCoords.**Append**([starting\_height - 1, starting\_width])

wallCoords.**Append**([starting\_height, starting\_width - 1])

wallCoords.**Append**([starting\_height, starting\_width + 1])

wallCoords.**Append**([starting\_height + 1, starting\_width])

maze[starting\_height-1][starting\_width] ← **False**

maze[starting\_height][starting\_width - 1] ← **False**

maze[starting\_height][starting\_width + 1] ← **False**

maze[starting\_height + 1][starting\_width] ← **False**

WHILE (wallCoords)

rand\_wall ← wallCoords[*Int*(***random()***\*wallCoords.*Length*)-1]

IF (rand\_wall[1] <> 0) THEN

IF (maze[rand\_wall[0]][rand\_wall[1]-1] = "None" AND

maze[rand\_wall[0]][rand\_wall[1]+1] = **True**) THEN

s\_cells = surroundingCells(maze, rand\_wall)

IF s\_cells < 2 THEN

maze[rand\_wall[0]][rand\_wall[1]] ← **True**

IF rand\_wall[0] <> 0 THEN

IF (maze[rand\_wall[0]-1][rand\_wall[1]] <> **True**) THEN

maze[rand\_wall[0]-1][rand\_wall[1]] ← **False**

ENDIF

IF ([rand\_wall[0]-1, rand\_wall[1]] NOT IN wallCoords) THEN

wallCoords.**Append**([rand\_wall[0]-1, rand\_wall[1]])

ENDIF

ENDIF

IF rand\_wall[0] <> MAZE\_HEIGHT-1 THEN

IF (maze[rand\_wall[0]+1][rand\_wall[1]] <> **True**) THEN

maze[rand\_wall[0]+1][rand\_wall[1]] ← **False**

ENDIF

IF ([rand\_wall[0]+1, rand\_wall[1]] NOT IN wallCoords) THEN

wallCoords.**Append**([rand\_wall[0]+1, rand\_wall[1]])

ENDIF

ENDIF

IF (rand\_wall[1] <> 0) THEN

IF (maze[rand\_wall[0]][rand\_wall[1]-1] <> **True**) THEN

maze[rand\_wall[0]][rand\_wall[1]-1] ← **False**

ENDIF

IF ([rand\_wall[0], rand\_wall[1]-1] NOT IN wallCoords) THEN

wallCoords.**Append**([rand\_wall[0], rand\_wall[1]-1])

ENDIF

ENDIF

ENDIF

FOR wall IN wallCoords

IF (wall[0] = rand\_wall[0] AND wall[1] = rand\_wall[1]) THEN

wallCoords.**Remove**(wall)

ENDIF

ENDFOR

ENDIF

CONTINUE

ENDIF

IF rand\_wall[0] <> 0 THEN

IF (maze[rand\_wall[0]-1][rand\_wall[1]] = "None" AND

maze[rand\_wall[0]+1][rand\_wall[1]] = **True**) THEN

s\_cells = surroundingCells(maze, rand\_wall)

IF s\_cells < 2 THEN

maze[rand\_wall[0]][rand\_wall[1]] ← **True**

IF rand\_wall[0] <> 0 THEN

IF (maze[rand\_wall[0]-1][rand\_wall[1]] <> **True**) THEN

maze[rand\_wall[0]-1][rand\_wall[1]] ← **False**

ENDIF

IF ([rand\_wall[0]-1, rand\_wall[1]] NOT IN wallCoords) THEN

wallCoords.Append([rand\_wall[0]-1, rand\_wall[1]])

ENDIF

ENDIF

IF rand\_wall[0] <> MAZE\_HEIGHT-1 THEN

IF (maze[rand\_wall[0]+1][rand\_wall[1]] <> **True**) THEN

maze[rand\_wall[0]+1][rand\_wall[1]] ← **False**

ENDIF

IF ([rand\_wall[0]+1, rand\_wall[1]] NOT IN wallCoords) THEN

wallCoords.Append([rand\_wall[0]+1, rand\_wall[1]])

ENDIF

ENDIF

IF (rand\_wall[1] <> 0) THEN

IF (maze[rand\_wall[0]][rand\_wall[1]-1] <> **True**) THEN

maze[rand\_wall[0]][rand\_wall[1]-1] ← **False**

ENDIF

IF ([rand\_wall[0], rand\_wall[1]-1] NOT IN wallCoords) THEN

wallCoords.Append([rand\_wall[0], rand\_wall[1]-1])

ENDIF

ENDIF

ENDIF

FOR wall IN wallCoords

IF (wall[0] = rand\_wall[0] AND wall[1] = rand\_wall[1]) THEN

wallCoords.**Remove**(wall)

ENDIF

ENDFOR

ENDIF

CONTINUE

ENDIF

IF rand\_wall[0] <> MAZE\_HEIGHT-1 THEN

IF (maze[rand\_wall[0]+1][rand\_wall[1]] = "None" AND

maze[rand\_wall[0]-1][rand\_wall[1]] = **True**) THEN

IF s\_cells < 2 THEN

maze[rand\_wall[0]][rand\_wall[1]] ← **True**

IF rand\_wall[0] <> 0 THEN

IF (maze[rand\_wall[0]-1][rand\_wall[1]] <> **True**) THEN

maze[rand\_wall[0]-1][rand\_wall[1]] ← **False**

ENDIF

IF ([rand\_wall[0]-1, rand\_wall[1]] NOT IN wallCoords) THEN

wallCoords.**Append**([rand\_wall[0]-1, rand\_wall[1]])

ENDIF

ENDIF

IF rand\_wall[0] <> MAZE\_HEIGHT-1 THEN

IF (maze[rand\_wall[0]+1][rand\_wall[1]] <> **True**) THEN

maze[rand\_wall[0]+1][rand\_wall[1]] ← **False**

ENDIF

IF ([rand\_wall[0]+1, rand\_wall[1]] NOT IN wallCoords) THEN

wallCoords.**Append**([rand\_wall[0]+1, rand\_wall[1]])

ENDIF

ENDIF

IF (rand\_wall[1] <> 0) THEN

IF (maze[rand\_wall[0]][rand\_wall[1]-1] <> **True**) THEN

maze[rand\_wall[0]][rand\_wall[1]-1] ← **False**

ENDIF

IF ([rand\_wall[0], rand\_wall[1]-1] NOT IN wallCoords) THEN

wallCoords.**Append**([rand\_wall[0], rand\_wall[1]-1])

ENDIF

ENDIF

ENDIF

FOR wall IN wallCoords

IF (wall[0] = rand\_wall[0] AND wall[1] = rand\_wall[1]) THEN

wallCoords.**Remove**(wall)

ENDIF

ENDFOR

ENDIF

CONTINUE

ENDIF

IF rand\_wall[1] <> MAZE\_WIDTH-1 THEN

IF (maze[rand\_wall[0]][rand\_wall[1]+1] = "None" AND

maze[rand\_wall[0]][rand\_wall[1]-1] = **True**) THEN

IF s\_cells < 2 THEN

maze[rand\_wall[0]][rand\_wall[1]] ← **True**

IF rand\_wall[0] <> 0 THEN

IF (maze[rand\_wall[0]-1][rand\_wall[1]] <> **True**) THEN

maze[rand\_wall[0]-1][rand\_wall[1]] ← **False**

ENDIF

IF ([rand\_wall[0]-1, rand\_wall[1]] NOT IN wallCoords) THEN

wallCoords.Append([rand\_wall[0]-1, rand\_wall[1]])

ENDIF

ENDIF

IF rand\_wall[0] <> MAZE\_HEIGHT-1 THEN

IF (maze[rand\_wall[0]+1][rand\_wall[1]] <> **True**) THEN

maze[rand\_wall[0]+1][rand\_wall[1]] ← **False**

ENDIF

IF ([rand\_wall[0]+1, rand\_wall[1]] NOT IN wallCoords) THEN

wallCoords.Append([rand\_wall[0]+1, rand\_wall[1]])

ENDIF

ENDIF

IF (rand\_wall[1] <> 0) THEN

IF (maze[rand\_wall[0]][rand\_wall[1]-1] <> **True**) THEN

maze[rand\_wall[0]][rand\_wall[1]-1] ← **False**

ENDIF

IF ([rand\_wall[0], rand\_wall[1]-1] NOT IN wallCoords) THEN

wallCoords.Append([rand\_wall[0], rand\_wall[1]-1])

ENDIF

ENDIF

ENDIF

FOR wall IN wallCoords

IF (wall[0] = rand\_wall[0] AND wall[1] = rand\_wall[1]) THEN

wallCoords.**Remove**(wall)

ENDIF

ENDFOR

ENDIF

CONTINUE

ENDIF

FOR wall IN wallCoords

IF (wall[0] = rand\_wall[0] AND wall[1] = rand\_wall[1]) THEN

wallCoords.**Remove**(wall)

ENDIF

ENDFOR

ENDWHILE

RETURN maze

ENDFUNCTION

PROCEDURE generate\_maze(MAZE\_HEIGHT, MAZE\_WIDTH, maze, walls, cells, all\_sprites)

FOR x = 0 TO MAZE\_HEIGHT-1

mazeRow = []

FOR y = 0 TO MAZE\_WIDTH - 1

mazeRow.**Append**("None")

ENDFOR

maze.**Append**(mazeRow)

ENDFOR

maze = create\_paths(maze, MAZE\_HEIGHT, MAZE\_WIDTH)

FOR i = 0 TO MAZE\_HEIGHT -1

FOR j = 0 TO MAZE\_WIDTH -1

IF maze[i][j] = "None" THEN

maze[i][j] ← **False**

ENDIF

maze[0][1] ← "None"

maze[MAZE\_HEIGHT-1][MAZE\_WIDTH-2] ← "None"

FOR x = 0 TO MAZE\_HEIGHT -1

FOR y = 0 TO MAZE\_WIDTH -1

X ← (25\*x)+500

Y ← (25\*y)+100

IF maze[x][y] = **False**

wall ← new Wall([X,Y])

walls.**Add**(wall)

all\_sprites.**Add**(wall)

ENDIF

ELSEIF maze[x][y] = **True**

cell ← new Cell([X,Y])

cells.**Add**(cell)

all\_sprites.**Add**(cell)

ENDELSEIF

ELSE

empty ← new Empty([X,Y])

all\_sprites.**Add**(empty)

ENDELSE

ENDFOR

ENDFOR

ENDPROCEDURE

CLASS pygame\_window

PRIVATE maze\_height

PRIVATE maze\_width

PRIVATE maze

PRIVATE win

PRIVATE player

PRIVATE all\_sprites

PRIVATE walls

PRIVATE cells

PUBLIC PROCEDURE new(MAZE\_HEIGHT, MAZE\_WIDTH, maze)

maze\_height ← MAZE\_HEIGHT

maze\_width ← MAZE\_WIDTH

maze ← maze

***pygame***.init()

win ← ***pygame***.display.set\_mode((1200, 600))

***pygame***.display.set\_caption("Maths Maze Navigator")

player ← new Player("Player.png",[500,100])

all\_sprites ← ***pygame***.sprite.Group()

walls ← ***pygame***.sprite.Group()

cells ← ***pygame***.sprite.Group()

all\_sprites.add(self.\_\_player)

ENDPROCEDURE

PUBLIC METHOD display

generate\_maze(maze\_height, maze\_width, maze, walls, cells, all\_sprites)

run ← **True**

WHILE run

FOR event in ***pygame***.event.get()

IF event.type = ***pygame***.QUIT

run ← **False**

ENDIF

ENDFOR

win.*fill*((0,0,0))

pressed\_keys ← ***pygame***.key.get\_pressed()

FOR sprite IN all\_sprites

win.*blit*(sprite.surf, sprite.rect)

ENDFOR

***pygame***.display.update()

player.move(pressed\_keys)

IF ***pygame***.sprite.spritecollideany(player, walls) THEN

player.*kill*()

run ← **False**

ENDIF

ENDWHILE

IF NOT run THEN

***sleep(2)***

***pygame***.quit()

ENDIF

ENDMETHOD

ENDCLASS

CLASS Game

PRIVATE display1

PUBLIC PROCEDURE new(MAZE\_HEIGHT, MAZE\_WIDTH, maze)

display1 ← new pygame\_window(MAZE\_HEIGHT, MAZE\_WIDTH, maze)

ENDPROCEDURE

PUBLIC METHOD run

display1.display()

ENDMETHOD

ENDCLASS

IF \_\_name\_\_ = "\_\_main\_\_" THEN

game = new Game(MAZE\_HEIGHT, MAZE\_WIDTH, maze)

game.run()

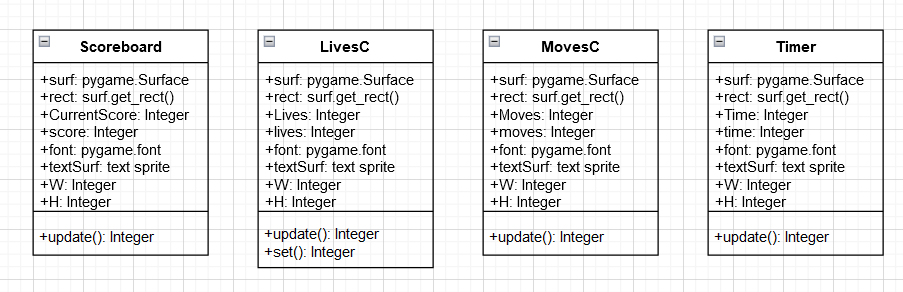
ENDIF

Information objects

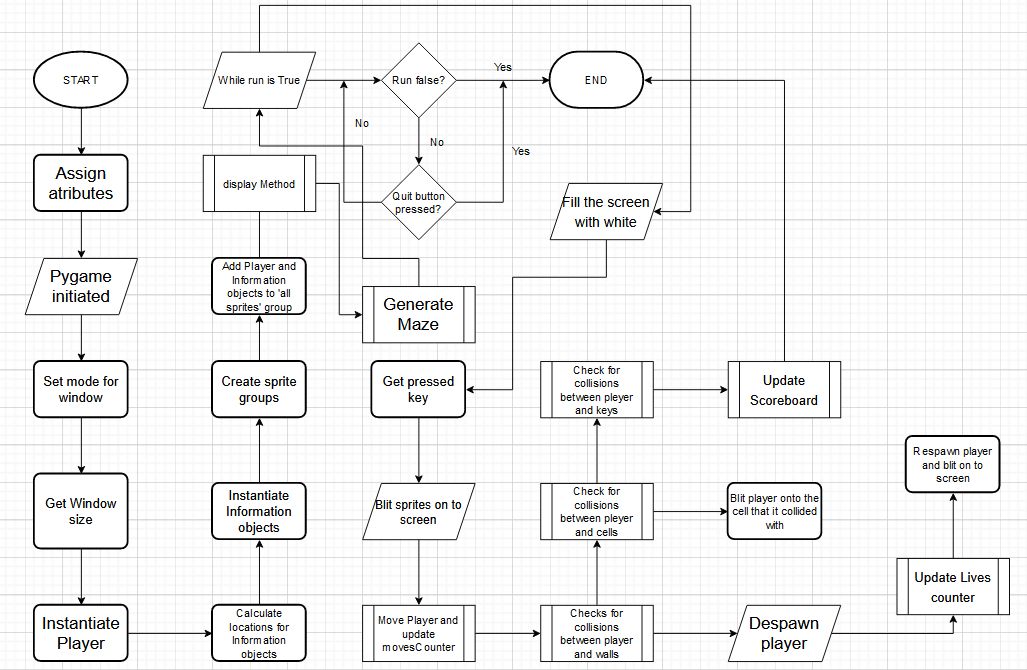
The next thing I am going to do is create the scoreboard, lives counter, moves counter and timer. The respective classes for these will be called Scoreboard, LivesC, MovesC and Timer

These will be done in a subprogram called ‘‘Information\_objects’

Below is the class diagram for this subprogram:



Below is the flowchart of the updated main program after the information objects have been added (only pygame\_window class):



The pseudocode for the ‘Information objects’ subprogram can be found in Appendix B3

On the next page, is part of the Pseudocode for the Main program after it has been updated