**GameMechanics.cpp**

GameMechanics place in Qwirkle is the ‘guard at the gates.’ - *The rules of the game* -

The approach to this class was to keep it as encapsulated as possible with minimal handling of memory. As such it’s role is not to change state or variable information but to simply offer answers to questions that were specific to the game’s rules.

At bare minimum the class offers three accessible methods

* **bool checkPosition()**

*Alerts gameplay whether or not a placement is legal*

* **bool isQwirkle()**

*Alerts gameplay if a Qwirkle has occurred*

* **int getPoints()**

*Returns gameplay all points gained in the single move*

**checkPosition** - *searched for all boundary conditions and local tiles to determine the legal action. Checking if tiles matched, position was available etc.*

**isQwirkle** - *looked if tiles lined up six in a row of legal placements.*

**getPoints** - *searched for all possible lines that existed in placement and utilised* ***isQwirkle*** *for any bonus points given.*

All methods take the same arguments, ***Tile to be place****,* ***Desired Position*** *and a* ***Placed Tiles List*** *of all tiles and their positions legally placed on the board.*

Some arguments utilised specific **typedefs** such as:

* PosPtr (a **pointer** to a **board position**, which also stored the set tile).
* PosVec (A vector of **board position** pointers)

**GameMechanics** *Enhancements:* **Get Hints**

* **PosPtr getHint()**

*Notify user of best possible tile to place and at which position*

Entering milestone 3 it was useful to utilise **GameMechanics** code to search all possible permutations between a players hand and tiles placed on board.

Reusing **GameMechanics** methods the method was able to find the optimum position and return a **PosPtr** → **BoardPosition** that held the position and tile.

**Test Cases**

Test cases run for GameMechanics were focused on **boundary** placements, **complicated** tile insertion and **non-repeatable** placements, within a line or adjacent moves.