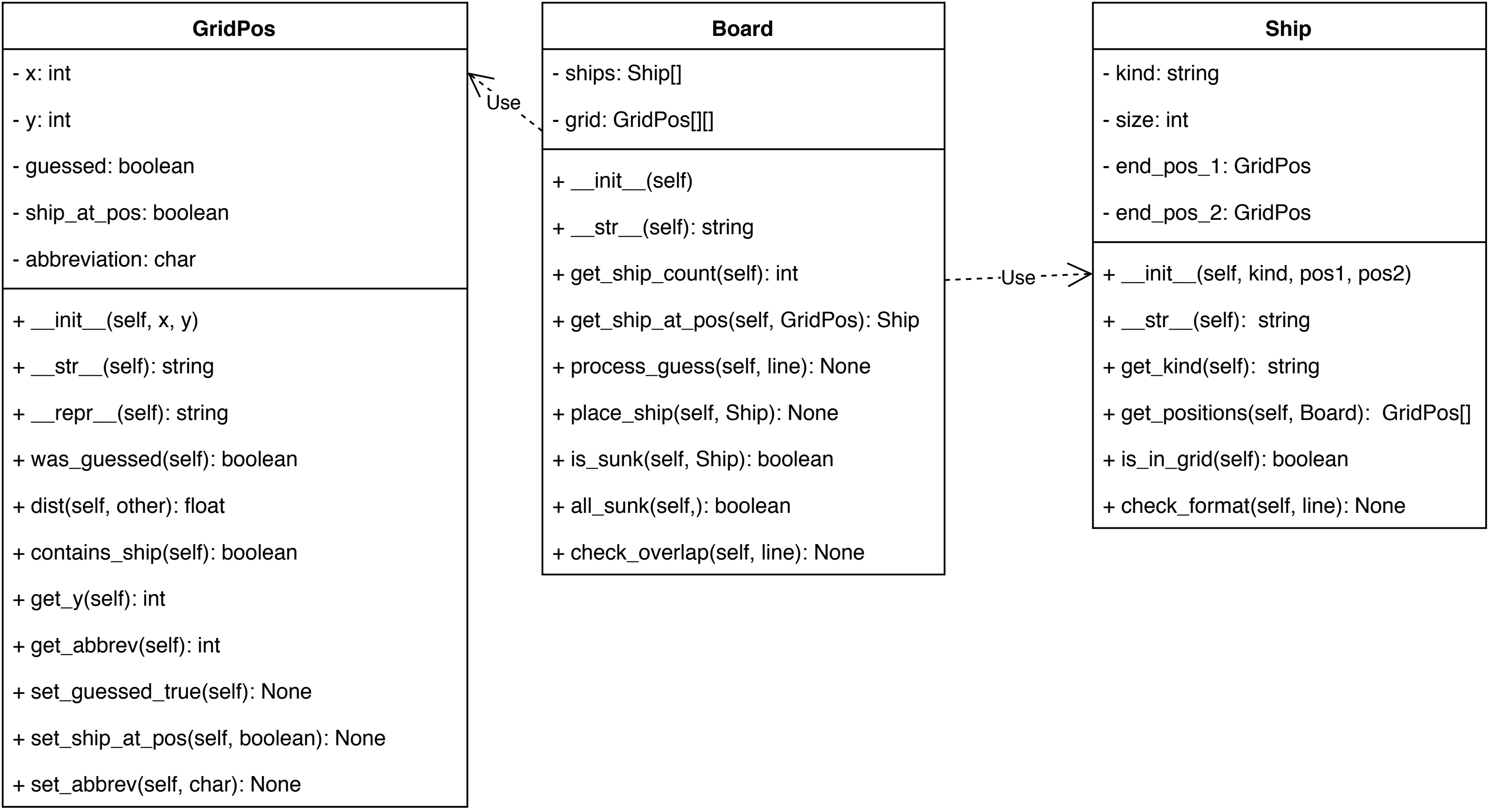

GridPos

- x: int
- y: int
- guessed: boolean
- ship_at_pos: boolean
- abbreviation: char

+ __init__(self, x, y)
+ __str__(self): string
+ __repr__(self): string
+ was_guessed(self): boolean
+ dist(self, other): float
+ contains_ship(self): boolean
+ get_y(self): int
+ get_abbrev(self): int
+ set_guessed_true(self): None
+ set_ship_at_pos(self, boolean): None
+ set_abbrev(self, char): None

Board

- ships: Ship[]
- grid: GridPos[][]

+ __init__(self)
+ __str__(self): string
+ get_ship_count(self): int
+ get_ship_at_pos(self, GridPos): Ship
+ process_guess(self, line): None
+ place_ship(self, Ship): None
+ is_sunk(self, Ship): boolean
+ all_sunk(self,): boolean
+ check_overlap(self, line): None

Ship

- kind: string
- size: int
- end_pos_1: GridPos
- end_pos_2: GridPos

+ __init__(self, kind, pos1, pos2)
+ __str__(self): string
+ get_kind(self): string
+ get_positions(self, Board): GridPos[]
+ is_in_grid(self): boolean
+ check_format(self, line): None

Main Execution

1. Create a new Board

File 1: Placement File

1. Open the file (making sure to use try/except for IOErrors)
2. Observe the file formatting:

B 3 2 3 5

The first value is the ship _kind
The second value is the first end of the ship's x_pos
The third value is the first end of the ship's y_pos
The fourth value is the other end of the ship's x_pos
The fifth value is the other end of the ship's y_pos

3. For each line in the file, we need to do two things:

1. Check the line! (using try/except as stated in the spec)
2. Create the Ship. This can be done by creating two GridPos objects, and then feeding the kind and the two GridPos's to the __init__ of our Ship class.
3. Add the newly created Ship to the Board

File 2: Guess File

1. Open the file (same story with the try/except)
2. Observe the file formatting:

0 1

The first value is an x_pos
The second value is a y_pos.
3. For each line in the file, check that the line is valid.
4. Split the line on whitespace.
5. Create a GridPos object from the resulting list of coordinates.
6. Query the board and see if that GridPos holds a ship. If so, you have a hit. If not, you missed. Make sure to take into account the possibility that the position had been guessed before.

The above diagram will be gone over in class on Wednesday (7/3/19). Please keep in mind that while this is a valid solution and will work for this assignment, there are some extras and possibly unnecessary methods included in the diagram to demonstrate design decision-making.