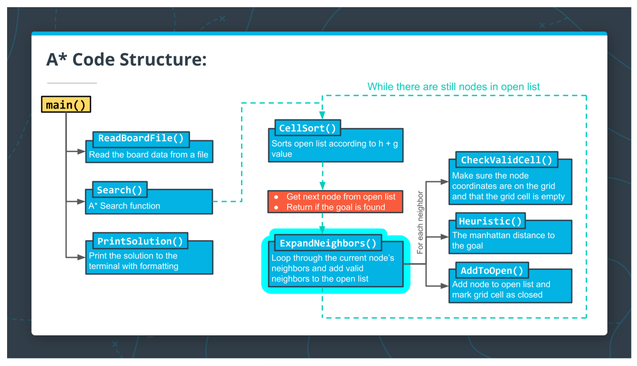
Expand the A* Search to Neighbors



Writing the `ExpandNeighbors()` function

You have now reached the final step of the A* algorithm! You are ready to expand your A* search to neighboring nodes and add valid neighbors to the open vector. In this exercise, you will write an `ExpandNeighbors` function that takes care of this functionality for you.

To Complete This Exercise:

> Write a `void ExpandNeighbors` function that accepts references to the following:
>
> - The current node,
> - the open vector,
> - the grid, and
> - an int array for the goal coordinates.
>
> The `ExpandNeighbors` function should implement the functionality given in the pseudocode below:

```
// TODO: ExpandNeighbors {

  // TODO: Get current node's data.

  // TODO: Loop through current node's potential neighbors.

    // TODO: Check that the potential neighbor's x2 and y2 values are on the grid and not closed.

      // TODO: Increment g value, compute h value, and add neighbor to open list.

// } TODO: End function
```

Note: we have provided directional deltas in the form of a 2D **array**. An array is a C++ container much like a vector, although without the ability to change size after initialization. Arrays can be accessed and iterated over just as vectors.

In the exercise, you can iterate over these `delta` values to check the neighbors in each direction:

```
// directional deltas
const int delta[4][2]{{-1, 0}, {0, -1}, {1, 0}, {0, 1}};
```