Write a While Loop for the A* Algorithm

Great work so far! Now on to some of the core functionality of the A* search algorithm. A* search works by sorting the open list using the f-value, and using the node with the lowest f-value as the next node in the search. This process continues until the goal node has been found or the open list runs out of nodes to use for searching.

In this exercise, you will implement the primary `while` loop in the algorithm which carries out the process described above:

To Complete This Exercise:

Complete all of the TODOs in the pseudocode below. These are also marked directly in the exercise code.

```
// TODO: while open vector is non empty {
  // TODO: Sort the open list using `CellSort`, and get the current node.

  // TODO: Get the x and y values from the current node,
  // and set grid[x][y] to kPath.

  // TODO: Check if you've reached the goal. If so, return grid.


  // If we're not done, expand search to current node's neighbors. This step will
  // ExpandNeighbors

//} // TODO: End while loop
```

Note: We've included a header and a function to sort the open vector:

- `#include <algorithm>`
- `std::sort`
- `CellSort`

The `CellSort` function uses the `Compare` function you wrote previously to determine the sorting order. The `CellSort` function contains two operators that you haven't seen