Occupancy grid documentation

This C++ program reads sensor data from a CSV file, simulates a robot's movement and sensor readings, and then creates an occupancy grid based on the data. The occupancy grid is a 2D representation of the environment where the robot moves, with '#' representing obstacles and '.' representing empty spaces. The program uses the FreeImage library to save the generated grid as a bitmap image.

Here is a breakdown of the main components:

1. Record Structure:
   - Represents a single row of sensor data.
   - Contains fields for timestamp, x and y positions, angle, and an array for ultrasound readings.

2. OccupancyGrid Class:
   - Manages the creation of an occupancy grid.
   - Reads data from a CSV file and stores it in a vector of `Record` structures.
   - Provides a method (`timeToDistance`) to convert ultrasound time readings to distances.
   - Defines a method (`updateGrid`) to update the occupancy grid based on robot movement and sensor readings.
   - Uses a method (`scale`) to transform real-world coordinates into grid coordinates.
   - Implements a method (`createMap`) to iterate through the recorded data, updating the grid and saving a bitmap image.
   - Includes methods (`printGrid`, `printRecords`) to print the grid and sensor data.

3. Main Function:
   - Creates an instance of the `OccupancyGrid` class.
   - Calls the `createMap` method to generate the occupancy grid based on the recorded data.
   - The resulting grid is printed to the console and saved as a bitmap image.

4. saveBitmap Method:
   - Uses the FreeImage library to create a bitmap image from the occupancy grid.
   - Iterates through the grid, setting pixel colors based on cell content ('#' for obstacles, '.' for empty spaces).
   - Saves the resulting bitmap image using the FreeImage library.

The program simulates a robot moving in a 2D space, updating an occupancy grid based on its position, orientation, and ultrasound sensor readings. The bitmap image visually represents the occupancy grid, helping visualize the robot's movement and the obstacles it encounters.

For further information or reference:
- FreeImage Library: https://freeimage.sourceforge.io/
- C++ Standard Library: https://en.cppreference.com/w/
- CSV File Reading in C++: https://www.geeksforgeeks.org/csv-file-reading-and-writing-in-c/