Code Jam 2011 - Round 1C

Analysis: Square Tiles

One solution for this problem would be to try to put red tiles in all possible ways over the blue tiles and see if at least one possibility leads to all the blue tiles being covered. However, this will not work in time because there are too many combinations to try.

To optimize the solution, you have to observe that if there is a solution, the top-most, left-most blue tile in the grid (that is, the left-most tile of all the blue tiles in the top row that contains any blue tiles) must be covered by the left-top corner of some red tile. This is because the tiles at its left and top are white (or non-existent), and so the red tile covering our blue tile cannot extend to the left or upwards of it. Based on this observation, we can solve the problem greedily by putting a red tile over the top-most, left-most blue tile in the only way it can be done. If for some blue tile it is impossible to cover it this way (because the red tile would cover some white tiles or extend outside the picture), then it's impossible to cover the whole board.

Note that as we are always sure that any red tile we put down is correct (if a solution exists at all), we can just modify the board on the fly, and thus at the same time check for solution existence and retrieve the answer. To illustrate that, here is a C++ function to cover all the blue tiles in a grid and return whether it was possible or not:

```
bool CoverTiles(vector<string>& grid) {
  const int m = (int)grid.size(), n = (int)grid[0].size();
  for (int i = 0; i < m; ++i)
    for (int j = 0; j < n; ++j)
      if (grid[i][j] == '#') {
        for (int di = 0; di < 2; ++di)
          for (int dj = 0; dj < 2; ++dj)
            if (i + di < m && j + dj < n &&
                grid[i + di][j + dj] == '#')
              grid[i + di][j + dj] = "/\\"[(di + dj) % 2];
            else
              return false;
      }
  return true;
}
```

Fun fact: the solution does not change if we drop the requirement that red tiles have to cover 2x2 squares of blue tiles - it is still valid if we allow the rotation and arbitrary positioning of the red tiles; the only condition that matters is that red tiles lie only on blue tiles (do not overlap, stick outside the picture or lie on white tiles) and all the blue tiles are covered in the end.