```
 1: /*
 2:  * queens.cc
 3:  *
 4:  *  Created on: Feb 14, 2017
 5:  *      Author: lubo
 6:  */
 7:
 8: #include <vector>
 9: #include <cstdlib>
10: #include <iostream>
11:
12: class QueensBoard {
13:     int size_;
14:     std::vector<int> board_;
15:
16: public:
17:     QueensBoard(int size)
18:             : size_(size), board_(size_, -1) {
19:     }
20:
21:     int size() const {
22:         return size_;
23:     }
24:
25:     bool under_attach(int row, int col) const {
26:         for (int i = 0; i < col; ++i) {
27:             if (board_[i] == -1) {
28:                 continue;
29:             }
30:             if (board_[i] == row) {
31:                 return true;
32:             }
33:             if (std::abs(i - col) == std::abs(board_[i] - row)) {
34:                 return true;
35:             }
36:         }
37:         return false;
38:     }
39:
40:     bool solve(int col = 0) {
41:         if (col == size()) {
42:             return true;
43:         }
44:         std::cout << "exploring col " << col << std::endl;
45:         for (int row = 0; row < size(); ++row) {
46:             if (!under_attach(row, col)) {
47:                 board_[col] = row;
48:                 std::cout << "placing queen on row " << row
49:                         << std::endl;
50:                 if (solve(col + 1)) {
51:                     return true;
52:                 }
53:             }
54:         }
55:         board_[col] = -1;
56:         return false;
57:     }
58:
59:     void pretty_print() const {
60:         std::cout << std::endl;
61:         for (int row = 0; row < size(); ++row) {
62:             for (int col = 0; col < size(); ++col) {
63:                 std::cout << '|';
64:                 if (board_[col] == row) {
65:                     std::cout << '*';
66:                 } else {
67:                     std::cout << ' ';
68:                 }
69:             }
70:             std::cout << '|' << std::endl;
71:         }
72:     }
73: };
74:
75: int main() {
76:
77:     QueensBoard qb(4);
78:     bool has_solution = qb.solve();
79:     std::cout << "has_solution=" << has_solution << std::endl;
80:     qb.pretty_print();
81:
82:     return 0;
83: }
84:
```

```
 1: #include <iostream>
 2: using namespace std;
 3:
 4: void b(int, int);
 5: void c(int, int);
 6: void d(int, int);
 7:
 8: void a(int i, int h) {
 9:
10:         if(i<=0) {
11:                 return;
12:         }
13:         d(i-1, h);
14:         cout << '-' << h << ' ' << 0 << ' ' << "rlineto" << endl;
15:         a(i-1, h);
16:         cout << 0 << ' ' << '-' << h << ' ' << "rlineto" << endl;
17:         a(i-1, h);
18:         cout << h << ' ' << 0 << ' ' << "rlineto" << endl;
19:         b(i-1, h);
20:
21: }
22:
23: void b(int i, int h) {
24:         if(i<=0) {
25:                 return;
26:         }
27:         c(i-1, h);
28:         cout << 0 << ' ' << h << ' ' << "rlineto" << endl;
29:         b(i-1, h);
30:         cout << h << ' ' << 0 << ' ' << "rlineto" << endl;
31:         b(i-1, h);
32:         cout << 0 << ' ' << '-' << h << ' ' << "rlineto" << endl;
33:         a(i-1, h);
34: }
35:
36: void c(int i, int h) {
37:         if(i<=0) {
38:                 return;
39:         }
40:         b(i-1, h);
41:         cout << h << ' ' << 0 << ' ' << "rlineto" << endl;
42:         c(i-1, h);
43:         cout << 0 << ' ' << h << ' ' << "rlineto" << endl;
44:         c(i-1, h);
45:         cout << '-' << h << ' ' << 0 << ' ' << "rlineto" << endl;
46:         d(i-1, h);
47: }
48:
49: void d(int i, int h) {
50:         if(i<=0) {
51:                 return;
52:         }
53:         a(i-1, h);
54:         cout << 0 << ' ' << '-' << h << ' ' << "rlineto" << endl;
55:         d(i-1, h);
56:         cout << '-' << h << ' ' << 0 << ' ' << "rlineto" << endl;
57:         d(i-1, h);
58:         cout << 0 << ' ' << h << ' ' << "rlineto" << endl;
59:         c(i-1, h);
60: }
61:
62:
63:
64: int main() {
65:         cout << "newpath" << endl;
66:         int h = 512;
67:         int x0 = 64 + h/2;
68:         int y0 = 64 + h/2;
69:         for(int i = 1; i<6; ++i) {
70:           h /=2;
71:           x0 += h/2;
72:           y0 += h/2;
73:           cout << x0 << " " << y0 << " moveto" << endl;
74:           a(i, h);
75:           // cout << i << " setlinewidth" << endl;
76:           cout << "stroke" << endl;
77:         }
78:         cout << "showpage" << endl;
79:
80:         return 0;
81: }
82:
```

```
    1: CXXFLAGS = -g -Wall
    2:
    3:
    4: OBJ = queens.o
    5: SRC = queens.cc hilbert.cc
    6:
    7: OUT = queens
    8:
    9:
   10: all: $(OUT)
   11:
   12:
   13: $(OUT): $(OBJ)
   14:         g++ $(CXXFLAGS) $(OBJ) -o $(OUT)
   15:
   16:
   17: clean:
   18:         rm -f *~ a.out $(OUT) *.o files.pdf
   19:
   20:
   21: files.pdf:
   22:         enscript -r -2 --highlight --line-numbers -o - $(SRC) Makefile | ps2pdf -
files.pdf
   23:
   24: pdf: files.pdf
```