

CISC181 Fall 2006 Project 1

Eight Queens

Submit electronically Thursday midnight. Deliver paper copy in class Friday.

Brute Force

Test systematically to find a solution. You should eliminate obvious non-solutions as soon as possible. I suggest placing one queen at a time, testing, then placing another (why?).

Random

Place a queen randomly, test (keeping it if it is good), place another. When should you give up searching at level n and “back up” to level $n - 1$? Do it randomly.

This method implies a degree of randomness, but more importantly it precludes completeness. In other words, if you have modified your brute force algorithm so that it operates in random order, that is still brute force.

If you place all eight queens randomly, test, and repeat, it might take a while. Be prepared to explain approximately how long.

Heuristic

A heuristic is a “rule of thumb”. It is an application of a guideline that is supposed to speed up an algorithm. For example, in a chess game one heuristic could be that the best next move is the move that offers the least chance that an opponent will capture a piece within n moves.

The heuristic I described in class was from Deitel & Deitel, a C++ text. Make a map of the board, and in each square s_i write the count of all squares that can be reached by a queen from s_i . The heuristic is that lower numbered squares are better picks than high ones.

Even though you are only required to find one solution with this method, it must be generally applicable, i.e. it must be *able* to find more than a single solution.

You may think up other heuristics or find them elsewhere and apply them. Cite the source of your heuristic if you didn't make it up.

Separate the Functions

This part is not about your script, is about how Bill will compare code.

Take your **fastest** method and place all necessary code in a “wrapper” function that will print the first board it finds and stop. Name it with your Strauss id and “fast”. Next, take your fastest (or only) function that will find **all** boards, and concatenate your user name with “all”.

For example, mine will be in functions:

```
void tharveyfast();
```

```
void tharveyall();
```

Place **both** of these functions in a file called ;username;.cc, e.g. tharvey.cc. This is so that Bill can place all of them in a larger program and race them. Winners get two slices at Peace or drink of choice at Brew Ha Ha.

Submission

Submit all your working code and script. Demonstrate your brute force method finding all solutions, and the other two each finding one solution.