N Queens

Bill Rubin

June 29, 2010

1. Implemented genetic algorithm in Visual C# to solve the 8 queens problem. Tuned the population size to 160 to optimize for this.
2. Generalized program for the N queens problem, for N up to 255.
3. Implemented multithreading solution, to fully exploit my Xeon 4-core processor. Separated the multithreading part from the N queens part, so that multithreading can be reused.
4. Tuned the population size to 3,000. (Smaller population size fails to converge for large N.) Larger population just takes more time.
5. Tuned reproduction percent to 80 and mutation percent to 60. Larger reproduction percents and/or mutation percents take more time; smaller ones seem less stable.
6. Found a solution for every size chessboard between 4 and 255 in less than 2.5 hours of computer time.

- **Approach:** Each of the N queens is constrained to be confined to one column, but can be in any of the N rows of that column. This reduces the problem to selecting which row each of the N queens should be in so that no queen attacks another. The genome is a N $\log_2$ N bit value: Each of the N queens requires $\log_2$ N bits to represent its position. There is a Genome class and a Population class. The program creates a Population object consisting of a number of randomly generated Genome objects. It then causes the Population object to iteratively breed.
- **How is the next generation produced?** (Final version) The 3,000 genomes are ranked in order of increasing (worsening) fitness value. The most-fit 48 percent breed; the rest are sterile. The most-fit 20 percent of the current generation survive into the next genration; the rest are killed. The most-fit 48 percent of the current generation are mutated; also, the most-fit 32 percent of the current generation are mated ("crossover") with their adjacent neighbors. In this way, the next generation ends up with the same size as the current generation. This scheme is much cruder (less random) than I'd originally hoped, but it worked anyway.
- **Breeding:** Mutation consists of flipping one bit of the Genome at random. Crossover consists of concatenating the first k bits of the first parent with the remaining bits of the second parent, k being chosen randomly.
- **Fitness Function:** The fitness function measures how many queens are on the same row or diagonal as each other. If queen 1 and queen 2 are on the same row or diagonal, this is counted exactly once. Therefore, the most fit Genome has fitness value zero and the least fit (all on same row, for example) has fitness value $N + (N-1) + (N-2) + ... + 1 = (N+1)N/2$. I know, high-fitness-bad is backwards from genetic algorithm tradition, but it seemed convenient to this problem.

- **Run-time Behavior:**   The algorithm converged on a solution for most, but not all, reasonable values of various parameters. However, the number of Genomes generated before convergence varied dramatically and unpredictably with even very small parameter changes. My genetic algorithm is ill-conditioned with respect to population size and mutation percent, but seems very robust with respect to how one selects parents to produce offspring.

- **Convergence:**   In every case I've observed, the minimum fitness of a population decreases monotonically, and fairly steadily, from some initial high number in the first generation down to one. From there, the minimum fitness often hangs there for a long time before converging (or until I lose patience at around 1,000 generations).

- **Sorting:**   In an early version, I fully sorted all the individuals in each generation, first by fitness, and next by genome value. Bad idea. This had the unfortunate effect that two genomes which begin with the same subsequence of rows were right next to each other. When they mated, they tended to produce children with the same genome as one parent. One ends up with lots (60 percent) of identical individuals in the population. I fixed this by not sorting by genome value. That way, individuals with the same fitness are ordered randomly.

- **Interbreeding:** Sometimes it can happen that the child of two parents is identical to one of the parents. If this occurs, I mutate that child. Otherwise, large numbers of duplicate individuals tend to occur. This slows or prevents convergence. Even with these mutations, one third of individuals tend to be duplicates. I don't quite understand why this is so.

- **Multithreading:** A Distributor object creates n worker threads, where n is the number of processors. Each thread waits for a task (Work). The main thread invokes the doWork() method to releases all threads from their wait state. The main thread then waits for all worker threads to complete their tasks. When a thread completes a task, the thread informs the waiting main thread of that fact, and then waits for more work. When the Distributor is asked to terminate the threads, it sets a bit which each worker thread uses to terminate itself.

- **Distributing the Work:** To create a new generation from the current one, each genome can be created independently of the others. The work is distributed evenly among the processors. A PopulationDistributor does this, using various kinds of work.

Output for 8 Queens

An early attempt, with population size optimized to 160.

"Rows:" for solution at bottom shows the row of each of the 8 queens (zero origin indexing).

```
Execution at 6/28/2010 8:00:55 AM

Genetic Algorithm Parameters:
        Population size:  160
        Reproduction percent:  80
        Mutation percent of reproduction:  30
        Mutation size:  38
        Crossover size:  90

Genome Parameter:
        Number of queens:       8
        Number of genome bits:  24
        Maximum fitness value:  28


                Fitness Function
                ------------------
Generation      Min     Mean    Max     Inbreeding
        1       3       7.6     14      0
        2       3       6.3     12      0
        3       2       5.3     11      1.88
        4       2       4.9     11      1.25
        5       1       4.6     10      0.63
        6       1       4.3     8       3.13
        7       1       4.2     8       4.38
        8       1       4.0     9       1.25
        9       1       3.8     11      2.5
        10      1       3.8     9       5
        11      1       3.7     9       7.5
        12      1       3.6     8       10
        13      1       3.5     10      8.75
        14      0       3.2     9       11.88

Fitness:  0
Rows:  7 3 0 2 5 1 6 4
Number of genomes created:  1,824

Total program elapsed time:  0.031 S
```

A later attempt, with population size set to 3,000, which works effectively for much larger number of queens.

"Rows:" for solution at bottom shows the row of each of the 8 queens (zero origin indexing).

```
Execution at 6/28/2010 8:01:33 AM

Genetic Algorithm Parameters:
        Population size:  3,000
        Reproduction percent:  80
        Mutation percent of reproduction:  30
        Mutation size:  720
        Crossover size:  1,680

Genome Parameter:
        Number of queens:        8
        Number of genome bits:  24
        Maximum fitness value:  28

                Fitness Function
                ------------------
Generation      Min     Mean    Max     Inbreeding
        1       1       7.8     18      0
        2       1       6.2     15      0.17
        3       1       5.4     14      0.8
        4       0       4.9     12      1.03

Fitness:  0
Rows:  6 3 1 4 7 0 2 5
Number of genomes created:  10,200

Total program elapsed time:  0.031 S
```

Output for 255 Queens

Execution at 6/28/2010 8:11:56 PM

Genetic Algorithm Parameters:
        Population size: 3,000
        Reproduction percent: 80
        Mutation percent of reproduction: 60
        Mutation size: 1,440
        Crossover size: 960

Genome Parameter:
        Number of queens: 255
        Number of genome bits: 2038.56
        Maximum fitness value: 32,385

```
                  Fitness Function
                 -------------------
Generation     Min     Mean     Max     Inbreeding
     1         245     296.2    356     0
     2         235     280.5    331     0
     3         234     271.4    330     0
     4         232     264.8    326     0.03
     5         232     259.2    309     0.03
     6         229     254.8    304     0.2
     7         219     250.9    314     0.27
     8         218     246.9    322     0.5
     9         213     242.6    299     0.8
    10         210     237.2    288     1.73

     ... (intermediate generations omitted)

    722         1       2.8      7      29.03
    723         1       2.8      6      29.53
    724         1       2.8      6      28.9
    725         0       2.9      7      28.4
```

Fitness: 0
Rows: 146 91 45 228 145 188 244 54 173 254 203 219 78 199 130 2 154 29 76 23
83 124 43 85 21 135 180 218 205 169 138 18 7 167 49 194 156 133 117 209 14
144 90 84 73 80 220 110 88 207 172 72 160 20 38 204 181 65 159 71 39 208 118
161 3 87 63 190 95 127 86 5 125 142 245 106 8 60 22 217 131 94 37 7 10 248 62
178 13 222 150 195 182 157 50 67 25 68 234 253 251 6 18 198 134 46 115 121 30
1 33 224 48 193 163 232 24 108 236 152 89 249 132 129 9 192 158 214 247 97
196 213 171 128 215 4 241 36 41 56 27 226 216 186 98 239 126 44 176 69 250
227 229 35 153 31 143 211 32 74 177 82 123 197 12 51 61 34 40 237 112 230 17
223 28 212 174 16 136 19 206 185 148 53 137 103 175 58 240 183 149 101 184
155 151 107 102 93 75 147 81 139 202 92 111 99 119 243 201 59 57 120 162 47
114 164 235 70 165 200 225 79 246 168 242 109 116 252 77 140 113 52 26 179 15
0 166 238 100 191 96 231 122 170 221 233 210 66 64 42 105 11 104 141 189 55
Number of genomes created: 1,740,600

Total program elapsed time: 168.828 S = 2.8 minutes

Note: Changing population size to 2,999 reduces effort for this case substantially: 454
generations, with **run time of 1.8 minutes**. See below.

Results for 1 to 150 Queens

Execution at 6/28/2010 5:44:01 PM

Genetic Algorithm Parameters:
    Population size: 3,000
    Reproduction percent: 80
    Mutation percent of reproduction: 60
    Mutation size: 1,440
    Crossover size: 960

| | Total Effort | | | Last Generation | | | | Effort Per Bit of Genome | | |
Number of Queens	Execution Time (S)	Number of Genomes	Number of Generations	Mean Fitness	Worst Fitness	Duplicate Genomes (percent)	Maximum Duplicate Genomes (percent)	Execution Time (mS)	Number of Genomes	Number of Generations
1	0.016	3,000	1	0.0	0	100.0	100.0	Infinity	Infinity	Infinity
2	--------- No solution found after 1,000 generations -----------------									
3	--------- No solution found after 1,000 generations -----------------									
4	0.015	3,000	1	3.3	6	91.5	91.5	1.9	375	0.125
5	0.000	3,000	1	4.4	10	35.2	35.2	0.0	258	0.086
6	0.000	5,400	2	4.3	11	4.8	4.8	0.0	348	0.129
7	0.016	10,200	4	4.1	11	2.3	2.3	0.8	519	0.204
8	0.016	15,000	6	4.1	11	2.0	2.0	0.7	625	0.250
9	0.015	12,600	5	5.1	12	1.2	1.2	0.5	442	0.175
10	0.031	31,800	13	4.3	13	2.0	2.0	0.9	957	0.391
11	0.079	58,200	24	3.8	12	10.4	10.4	2.1	1,529	0.631
12	0.062	58,200	24	4.2	12	5.1	5.1	1.4	1,353	0.558
13	0.063	46,200	19	5.1	15	1.8	1.9	1.3	960	0.395
14	0.093	75,000	31	4.5	14	6.9	6.9	1.7	1,407	0.582
15	0.110	75,000	31	3.4	12	24.6	24.6	1.9	1,280	0.529
16	0.125	87,000	36	3.0	7	36.0	36.0	2.0	1,359	0.563
17	0.125	89,400	37	4.7	17	9.3	9.3	1.8	1,287	0.532
18	0.109	67,800	28	3.3	8	27.3	28.8	1.5	903	0.373
19	0.125	84,600	35	3.5	13	27.8	28.2	1.5	1,048	0.434
20	0.188	111,000	46	3.1	13	43.8	45.9	2.2	1,284	0.532
21	0.296	168,600	70	2.9	7	39.5	39.8	3.2	1,828	0.759
22	0.157	87,000	36	3.9	8	19.9	19.9	1.6	887	0.367
23	0.797	387,000	161	2.8	8	41.5	42.9	7.7	3,720	1.547
24	0.188	99,000	41	3.3	7	27.6	27.7	1.7	900	0.373
25	0.297	147,000	61	2.8	7	32.3	32.3	2.6	1,266	0.525
26	0.656	291,000	121	2.7	7	38.6	38.6	5.4	2,381	0.990
27	0.265	123,000	51	3.8	15	27.7	27.7	2.1	958	0.397

28	0.641	262,200	109	2.8	7	32.0	33.2	4.8	1,948	0.810
29	0.344	147,000	61	3.0	7	30.6	30.6	2.4	1,043	0.433
30	0.359	142,200	59	3.0	7	28.7	33.7	2.4	966	0.401
31	0.719	252,600	105	3.1	8	31.3	32.4	4.7	1,645	0.684
32	1.094	387,000	161	2.7	7	35.2	35.2	6.8	2,419	1.006
33	0.375	137,400	57	4.0	8	19.3	19.3	2.3	825	0.342
34	0.343	125,400	52	3.5	8	27.4	27.4	2.0	725	0.301
35	0.516	175,800	73	3.6	8	22.7	25.8	2.9	979	0.407
36	0.719	223,800	93	2.9	7	31.1	31.4	3.9	1,202	0.500
37	0.625	190,200	79	3.1	7	28.9	28.9	3.2	987	0.410
38	0.594	175,800	73	3.5	8	28.0	28.0	3.0	882	0.366
39	0.828	233,400	97	2.4	6	41.4	41.4	4.0	1,132	0.471
40	0.734	209,400	87	3.1	7	29.9	32.2	3.4	984	0.409
41	0.844	231,000	96	3.1	7	25.8	26.4	3.8	1,052	0.437
42	0.719	190,200	79	3.8	8	24.5	27.2	3.2	840	0.349
43	0.921	235,800	98	2.8	7	31.7	32.0	3.9	1,011	0.420
44	3.235	756,600	315	2.4	6	39.6	39.7	13.5	3,150	1.311
45	0.750	180,600	75	3.0	7	30.1	30.1	3.0	731	0.303
46	1.031	238,200	99	3.8	8	24.5	27.3	4.1	937	0.390
47	1.031	231,000	96	3.0	7	25.5	32.3	3.9	885	0.368
48	1.157	257,400	107	3.6	8	20.3	23.3	4.3	960	0.399
49	2.125	427,800	178	2.6	7	36.2	37.1	7.7	1,555	0.647
50	1.359	281,400	117	3.5	7	27.0	27.0	4.8	997	0.415
51	1.250	247,800	103	3.3	7	30.1	32.2	4.3	857	0.356
52	1.875	348,600	145	3.1	7	27.1	27.9	6.3	1,176	0.489
53	4.609	783,000	326	2.4	6	36.8	39.6	15.2	2,579	1.074
54	2.266	384,600	160	3.0	7	27.2	27.7	7.3	1,238	0.515
55	0.937	171,000	71	4.1	8	24.3	24.3	2.9	538	0.223
56	1.360	228,600	95	3.1	7	24.2	25.6	4.2	703	0.292
57	1.312	226,200	94	3.5	8	23.0	26.6	3.9	680	0.283
58	2.094	329,400	137	3.0	7	28.4	33.0	6.2	969	0.403
59	2.672	394,200	164	2.6	7	32.6	33.7	7.7	1,136	0.473
60	2.359	358,200	149	2.9	7	28.8	37.4	6.7	1,011	0.420
61	1.438	214,200	89	3.2	7	24.8	24.8	4.0	592	0.246
62	2.297	329,400	137	3.0	7	25.3	32.7	6.2	892	0.371
63	1.344	190,200	79	3.2	7	25.6	25.6	3.6	505	0.210
64	4.172	531,000	221	2.8	7	28.1	28.5	10.9	1,383	0.576
65	1.859	250,200	104	3.5	8	26.0	26.0	4.7	639	0.266
66	3.125	399,000	166	2.7	7	32.8	32.9	7.8	1,000	0.416
67	1.953	247,800	103	3.8	8	27.4	27.5	4.8	610	0.253
68	3.204	382,200	159	3.0	7	32.4	33.3	7.7	923	0.384
69	2.859	348,600	145	2.7	7	33.2	33.3	6.8	827	0.344
70	7.000	775,800	323	2.7	7	30.1	31.5	16.3	1,808	0.753

71	5.453	600,600	250	2.5	7	34.7	35.8	12.5	1,376	0.573
72	2.813	310,200	129	3.2	7	26.2	26.2	6.3	698	0.290
73	2.625	293,400	122	3.9	8	26.8	27.5	5.8	649	0.270
74	3.671	394,200	164	2.4	6	40.2	41.4	8.0	858	0.357
75	3.704	365,400	152	3.0	7	27.1	27.7	7.9	782	0.325
76	3.109	315,000	131	3.3	7	23.7	26.6	6.5	663	0.276
77	3.922	384,600	160	2.8	7	31.1	31.9	8.1	797	0.332
78	6.094	552,600	230	3.1	7	26.5	28.7	12.4	1,127	0.469
79	4.390	415,800	173	2.8	7	30.6	32.6	8.8	835	0.347
80	3.719	341,400	142	3.4	7	26.5	29.0	7.4	675	0.281
81	5.344	471,000	196	2.5	7	35.6	35.6	10.4	917	0.382
82	5.078	459,000	191	3.1	7	25.2	30.8	9.7	880	0.366
83	4.781	411,000	171	2.6	6	36.0	36.3	9.0	777	0.323
84	4.703	375,000	156	3.8	8	24.2	25.2	8.8	698	0.291
85	4.328	346,200	144	3.2	7	24.2	24.8	7.9	635	0.264
86	4.156	351,000	146	2.5	6	36.3	37.5	7.5	635	0.264
87	11.578	895,800	373	2.4	6	40.1	41.6	20.7	1,598	0.665
88	3.641	293,400	122	3.5	8	27.3	27.3	6.4	516	0.215
89	5.047	396,600	165	3.3	7	26.9	31.3	8.8	688	0.286
90	5.062	379,800	158	3.1	7	22.9	31.3	8.7	650	0.270
91	6.282	449,400	187	3.0	7	24.9	28.9	10.6	759	0.316
92	4.625	334,200	139	3.5	8	25.6	26.5	7.7	557	0.232
93	4.110	303,000	126	3.7	8	23.7	27.2	6.8	498	0.207
94	6.546	444,600	185	3.2	7	28.8	33.0	10.6	722	0.300
95	14.079	831,000	346	2.8	7	28.6	30.0	22.6	1,331	0.554
96	16.046	1,061,400	442	2.6	7	33.6	37.0	25.4	1,679	0.699
97	8.125	509,400	212	3.3	7	24.2	31.6	12.7	796	0.331
98	11.594	732,600	305	2.3	5	40.2	41.9	17.9	1,130	0.471
99	5.000	324,600	135	3.7	8	24.9	27.8	7.6	495	0.206
100	12.391	744,600	310	2.5	6	38.1	38.9	18.7	1,121	0.467
101	14.000	835,800	348	2.6	7	33.9	33.9	20.8	1,243	0.517
102	7.078	411,000	171	3.1	7	23.3	30.2	10.4	604	0.251
103	6.281	379,800	158	3.0	7	28.8	28.8	9.1	551	0.229
104	10.797	591,000	246	2.7	7	30.8	31.9	15.5	848	0.353
105	5.219	310,200	129	3.9	8	25.2	27.9	7.4	440	0.183
106	6.906	387,000	161	3.5	8	29.8	32.3	9.7	543	0.226
107	8.000	435,000	181	3.3	7	24.3	28.0	11.1	603	0.251
108	12.422	639,000	266	3.0	7	25.9	29.0	17.0	876	0.365
109	13.750	720,600	300	3.1	7	29.2	31.5	18.6	977	0.407
110	13.391	679,800	283	2.5	6	34.5	34.9	18.0	911	0.379
111	21.265	982,200	409	3.3	7	22.6	24.0	28.2	1,302	0.542
112	13.469	658,200	274	2.7	7	30.4	30.7	17.7	863	0.359
113	19.922	922,200	384	2.7	7	31.5	31.5	25.8	1,197	0.498

114	9.469	451,800	188	2.9	7	28.0	29.6	12.2	580	0.241
115	11.421	567,000	236	2.6	6	34.7	35.0	14.5	720	0.300
116	8.719	418,200	174	3.0	7	27.7	28.5	11.0	526	0.219
117	13.094	598,200	249	2.8	7	29.8	31.3	16.3	744	0.310
118	19.109	845,400	352	3.0	7	25.3	31.3	23.5	1,041	0.433
119	13.485	603,000	251	3.0	7	25.9	31.1	16.4	735	0.306
120	12.468	543,000	226	3.1	7	24.6	27.1	15.0	655	0.273
121	11.079	487,800	203	3.2	7	23.2	31.8	13.2	583	0.242
122	14.453	615,000	256	2.6	7	34.4	34.5	17.1	727	0.303
123	16.593	672,600	280	2.9	7	31.9	35.0	19.4	788	0.328
124	15.297	622,200	259	2.7	7	32.3	32.3	17.7	722	0.300
125	16.032	639,000	266	2.9	6	29.4	31.6	18.4	734	0.305
126	28.375	1,140,600	475	2.6	7	31.2	35.2	32.3	1,297	0.540
127	12.812	509,400	212	3.3	7	26.4	28.9	14.4	574	0.239
128	11.906	480,600	200	3.0	7	24.5	32.6	13.3	536	0.223
129	21.328	833,400	347	2.6	7	35.8	36.6	23.6	921	0.384
130	21.157	780,600	325	2.4	6	36.5	39.3	23.2	855	0.356
131	12.297	475,800	198	3.4	7	32.5	32.6	13.3	516	0.215
132	11.796	449,400	187	3.0	6	25.1	31.4	12.7	483	0.201
133	29.469	1,051,800	438	2.5	7	34.2	35.3	31.4	1,121	0.467
134	19.953	703,800	293	3.0	7	25.6	30.7	21.1	743	0.309
135	22.438	785,400	327	2.5	7	36.7	38.5	23.5	822	0.342
136	26.281	893,400	372	2.9	7	25.7	30.2	27.3	927	0.386
137	23.203	795,000	331	2.5	6	36.9	37.5	23.9	818	0.340
138	14.672	502,200	209	3.1	7	23.8	30.4	15.0	512	0.213
139	18.953	627,000	261	3.0	7	26.1	29.1	19.2	634	0.264
140	15.453	509,400	212	2.5	7	38.0	39.3	15.5	510	0.212
141	29.453	936,600	390	2.9	7	27.5	33.8	29.3	930	0.387
142	24.297	756,600	315	2.4	7	36.9	36.9	23.9	745	0.310
143	23.000	643,800	268	2.9	7	29.8	34.0	22.5	629	0.262
144	33.594	1,063,800	443	2.8	7	28.1	33.5	32.5	1,030	0.429
145	20.188	619,800	258	3.1	7	26.6	33.7	19.4	595	0.248
146	21.750	667,800	278	2.7	7	32.5	33.3	20.7	636	0.265
147	39.203	1,128,600	470	3.0	7	25.4	30.2	37.0	1,066	0.444
148	28.468	831,000	346	2.5	7	35.9	35.9	26.7	779	0.324
149	30.704	876,600	365	2.4	5	37.0	39.4	28.5	815	0.339
150	29.656	845,400	352	2.6	7	34.9	35.8	27.3	780	0.325

Total program elapsed time: 1,145.36 S = 19.1 minutes

Results for 151 to 200 Queens

Execution at 6/29/2010 6:34:00 AM

Genetic Algorithm Parameters:
     Population size: 3,000
     Reproduction percent: 80
     Mutation percent of reproduction: 60
     Mutation size: 1,440
     Crossover size: 960

| | Total Effort | | | Last Generation | | | Maximum | Effort Per Bit of Genome | | |
Number of Queens	Execution Time (S)	Number of Genomes	Number of Generations	Mean Fitness	Worst Fitness	Duplicate Genomes (percent)	Duplicate Genomes (percent)	Execution Time (mS)	Number of Genomes	Number of Generations
151	23.157	691,800	288	3.1	7	28.0	33.3	21.2	633	0.263
152	33.203	941,400	392	3.0	7	27.2	33.7	30.1	855	0.356
153	19.532	562,200	234	2.7	7	33.0	33.0	17.6	506	0.211
154	30.828	821,400	342	2.6	7	35.1	35.1	27.5	734	0.306
155	29.547	831,000	346	3.0	7	27.0	33.1	26.2	737	0.307
156	34.375	876,600	365	2.8	7	30.3	31.2	30.2	771	0.321
157	31.500	869,400	362	2.4	5	39.2	39.5	27.5	759	0.316
158	24.328	636,600	265	3.2	7	28.2	34.4	21.1	552	0.230
159	34.844	879,000	366	2.5	6	35.3	37.5	30.0	756	0.315
160	34.500	850,200	354	3.0	7	27.1	34.9	29.4	726	0.302
161	34.484	855,000	356	2.4	6	37.1	37.5	29.2	724	0.302
162	21.969	559,800	233	2.9	6	30.1	30.1	18.5	471	0.196
163	53.734	1,378,200	574	2.4	6	36.4	38.3	44.9	1,151	0.479
164	27.906	677,400	282	2.4	5	39.5	39.6	23.1	561	0.234
165	49.453	1,227,000	511	2.3	6	40.0	40.6	40.7	1,010	0.420
166	54.985	1,308,600	545	2.5	6	34.0	38.6	44.9	1,069	0.445
167	22.906	579,000	241	3.2	7	24.5	33.9	18.6	470	0.195
168	41.984	948,600	395	2.7	6	32.2	33.7	33.8	764	0.318
169	47.610	1,071,000	446	2.5	7	36.1	36.8	38.1	856	0.357
170	66.500	1,546,200	644	3.0	7	30.1	33.7	52.8	1,228	0.511
171	39.344	903,000	376	2.8	7	31.1	35.9	31.0	712	0.296
172	56.500	1,234,200	514	2.4	5	34.8	37.4	44.2	966	0.402
173	54.421	1,159,800	483	2.4	6	33.7	36.9	42.3	902	0.376
174	36.110	797,400	332	2.3	6	40.8	40.8	27.9	616	0.256
175	44.406	931,800	388	3.1	7	24.8	30.4	34.1	715	0.298
176	33.797	713,400	297	3.1	7	22.9	30.6	25.7	543	0.226

```
177    45.141    958,200    399    2.5    6    35.4    36.2    34.2     725    0.302
178    34.000    732,600    305    3.4    7    30.0    32.8    25.6     551    0.229
179    46.531    951,000    396    3.0    7    27.4    31.7    34.7     710    0.296
180    53.109  1,119,000    466    3.0    7    26.0    33.1    39.4     830    0.346
181   109.375  2,081,400    867    2.7    7    33.3    33.3    80.6   1,533    0.639
182    50.375    977,400    407    2.5    7    35.7    36.5    36.9     715    0.298
183    49.219    946,200    394    2.6    7    34.6    36.2    35.8     688    0.286
184    50.750    970,200    404    3.1    8    25.2    33.2    36.7     701    0.292
185    45.547    907,800    378    2.7    7    32.8    34.3    32.7     652    0.271
186    43.609    809,400    337    2.7    6    33.1    35.6    31.1     577    0.240
187    40.469    797,400    332    2.8    7    31.6    35.0    28.7     565    0.235
188    33.766    655,800    273    3.3    7    28.7    31.3    23.8     462    0.192
189    47.171    910,200    379    2.7    7    32.6    34.4    33.0     637    0.265
190    36.719    706,200    294    3.4    7    27.6    30.2    25.5     491    0.204
191    38.985    723,000    301    3.5    7    24.1    31.5    26.9     500    0.208
192    59.656  1,068,600    445    2.6    7    34.0    35.0    41.0     734    0.306
193    37.469    696,600    290    3.1    7    27.3    34.0    25.6     475    0.198
194    72.546  1,308,600    545    2.2    6    40.5    43.5    49.2     888    0.370
195    50.782    886,200    369    3.1    6    23.5    34.5    34.2     597    0.249
196    52.359    929,400    387    2.3    5    39.5    41.3    35.1     623    0.259
197    71.891  1,174,200    489    2.8    7    31.3    31.8    47.9     782    0.326
198    73.172  1,195,800    498    3.1    7    24.2    32.3    48.4     792    0.330
199    98.312  1,637,400    682    2.5    7    35.1    37.5    64.7   1,077    0.449
200    57.062    929,400    387    2.4    7    38.3    40.7    37.3     608    0.253
```

Total program elapsed time: 2,279.906 S = 38.0 minutes

Results for 201 to 255 Queens

Execution at 6/28/2010 8:59:17 PM

Genetic Algorithm Parameters:
       Population size: 3,000
       Reproduction percent: 80
       Mutation percent of reproduction: 60
       Mutation size: 1,440
       Crossover size: 960

| | Total Effort | | | Last Generation | | | Maximum | Effort Per Bit of Genome | | |
Number of Queens	Execution Time (S)	Number of Genomes	Number of Generations	Mean Fitness	Worst Fitness	Duplicate Genomes (percent)	Duplicate Genomes (percent)	Execution Time (mS)	Number of Genomes	Number of Generations
201	38.422	643,800	268	3.6	8	27.8	30.4	25.0	419	0.174
202	68.141	1,109,400	462	2.5	7	34.8	36.9	44.0	717	0.299
203	78.406	1,191,000	496	3.0	6	26.3	29.9	50.4	765	0.319
204	56.625	941,400	392	2.9	7	31.2	36.8	36.2	601	0.250
205	64.281	1,025,400	427	3.1	7	25.0	33.9	40.8	651	0.271
206	50.235	811,800	338	3.3	7	24.8	31.6	31.7	513	0.213
207	82.890	1,279,800	533	2.3	6	39.4	39.8	52.0	804	0.335
208	95.063	1,469,400	612	2.2	5	42.5	44.1	59.4	917	0.382
209	68.172	1,042,200	434	3.0	6	25.0	33.8	42.3	647	0.269
210	73.265	1,099,800	458	2.7	7	32.2	34.9	45.2	679	0.283
211	--------- No solution found after 1,000 generations -----------------									
212	123.937	1,810,200	754	2.5	7	33.3	35.3	75.6	1,105	0.460
213	57.203	857,400	357	3.3	7	27.8	34.5	34.7	520	0.217
214	104.844	1,560,600	650	2.4	6	35.9	37.8	63.3	942	0.392
215	67.953	982,200	409	2.8	6	32.3	35.3	40.8	590	0.246
216	68.813	1,011,000	421	2.4	6	38.1	38.8	41.1	604	0.251
217	78.656	1,071,000	446	3.2	7	22.0	33.5	46.7	636	0.265
218	62.266	907,800	378	3.3	7	25.8	34.7	36.8	536	0.223
219	62.343	876,600	365	3.1	6	24.4	29.9	36.6	515	0.214
220	53.063	754,200	314	3.5	8	23.2	35.9	31.0	441	0.183
221	81.375	1,123,800	468	2.5	6	37.2	38.3	47.3	653	0.272
222	82.250	1,140,600	475	2.4	6	35.2	36.5	47.5	659	0.275
223	92.421	1,243,800	518	2.6	7	33.5	34.0	53.1	715	0.298
224	67.954	915,000	381	3.4	7	23.5	37.1	38.9	523	0.218
225	53.656	730,200	304	3.4	7	29.2	32.2	30.5	415	0.173
226	92.000	1,239,000	516	3.3	7	24.2	33.3	52.1	701	0.292
227	85.719	1,114,200	464	2.7	7	31.8	33.8	48.2	627	0.261

```
228   102.125 1,337,400    557    2.6    7    35.7    35.9    57.2     749    0.312
229    79.625 1,030,200    429    2.8    7    30.1    34.0    44.4     574    0.239
230   132.296 1,673,400    697    2.5    7    33.6    35.8    73.3     927    0.386
231    93.907 1,159,800    483    3.2    6    23.1    36.1    51.8     639    0.266
232   106.468 1,270,200    529    3.0    7    26.0    28.0    58.4     697    0.290
233    98.579 1,258,200    524    2.4    5    38.9    41.1    53.8     687    0.286
234    94.656 1,126,200    469    2.5    6    38.0    38.0    51.4     612    0.255
235   138.125 1,584,600    660    2.7    6    32.1    33.2    74.6     856    0.357
236   130.344 1,522,200    634    2.5    7    36.7    37.2    70.1     818    0.341
237   121.375 1,488,600    620    2.5    7    36.2    38.0    64.9     796    0.332
238    92.859 1,126,200    469    2.4    6    35.5    36.8    49.4     599    0.250
239   110.406 1,301,400    542    2.4    6    36.4    38.2    58.5     689    0.287
240    98.125 1,159,800    483    2.7    7    32.9    35.6    51.7     611    0.255
241   147.860 1,702,200    709    2.3    5    40.1    41.1    77.5     893    0.372
242    82.640   922,200    384    3.0    6    27.3    29.0    43.1     481    0.200
243   194.094 2,316,600    965    2.3    5    35.9    39.2   100.8   1,203    0.501
244   108.422 1,217,400    507    3.1    7    25.6    35.1    56.0     629    0.262
245   106.562 1,193,400    497    2.4    6    38.3    39.2    54.8     614    0.256
246    78.703   893,400    372    2.9    7    29.3    37.2    40.3     457    0.190
247    90.969 1,042,200    434    3.5    7    33.4    36.9    46.3     531    0.221
248   125.156 1,361,400    567    2.7    6    32.8    34.1    63.4     690    0.287
249   117.297 1,251,000    521    2.5    6    35.6    36.4    59.2     631    0.263
250    92.016 1,018,200    424    3.2    7    28.8    35.2    46.2     511    0.213
251   102.719 1,078,200    449    2.6    6    32.7    33.0    51.3     539    0.224
252   104.968 1,150,200    479    2.6    7    36.4    36.8    52.2     572    0.238
253   106.282 1,114,200    464    2.8    7    29.8    36.2    52.6     552    0.230
254   112.828 1,212,600    505    2.3    5    38.1    39.8    55.6     598    0.249
255   169.609 1,740,600    725    2.9    7    28.4    37.5    83.2     854    0.356

Total program elapsed time:   5,227.484 S = 87.1 minutes
```

Total effort for 1 through 255 queens: 2.4 hours

Anomaly at 211 Queens

The only failure to converge to a solution within 1,000 generations occurred for 211 queens. (A few other cases barely made it under the wire: 243 queens took 965 generations; 181 queens took 867 generations.) By expanding the generation limit, I found that 211 queens converged to a solution at 1,126 generations. Amazingly, the population minimum fitness reached 1 at generation 272, *and stayed there for 854 more generations*. Population mean and max fitness reached 3.3 and 7, respectively, at generation 313, and stayed near there until the end.

```
Execution at 6/29/2010 9:38:55 AM

Genetic Algorithm Parameters:
      Population size:  3,000
      Reproduction percent:  80
      Mutation percent of reproduction:  60
      Mutation size:  1,440
      Crossover size:  960
```

	Total Effort				Last Generation				Effort Per Bit of Genome		
		Number	Number			Duplicate	Maximum Duplicate		Number	Number
Number of Queens	Execution Time (S)	of Genomes	of Generations	Mean Fitness	Worst Fitness	Genomes (percent)	Genomes (percent)	Execution Time (mS)	of Genomes	of Generations
--------	-----------	--------	-----------	--------	--------	-----------	-----------	-----------	--------	-----------
211	202.594	2,703,000	1,126	3.3	7	21.7	29.6	124.4	1,659	0.691

```
Total program elapsed time:  202.641 S = 3.4 minutes
```

Small Parameter Changes

I've often observed that a run's population minimum fitness may get to 1 without difficulty, and then linger there a very long time, before getting to zero. A smarter algorithm might detect this case, and restart the search with different parameters. Tests show that even slightly different parameters have a substantial effect on all runs. For example, just decreasing the population size from 3,000 to 2,999 allows the 211 case to converge in just 486 generations, which is below the average for number of queens in that range:

```
Execution at 6/29/2010 9:44:39 AM

Genetic Algorithm Parameters:
        Population size:  2,999
        Reproduction percent:  80
        Mutation percent of reproduction:  60
        Mutation size:  1,439
        Crossover size:  960
```

	Total Effort			Last Generation			Maximum	Effort Per Bit of Genome		
Number of Queens	Execution Time (S)	Number of Genomes	Number of Generations	Mean Fitness	Worst Fitness	Duplicate Genomes (percent)	Duplicate Genomes (percent)	Execution Time (mS)	Number of Genomes	Number of Generations
211	73.937	1,166,514	486	2.3	5	39.1	41.1	45.4	716	0.298
250	84.968	933,811	389	2.7	7	34.0	34.2	42.7	469	0.195
251	129.407	1,399,217	583	2.2	5	41.1	42.4	64.7	699	0.291
252	110.484	1,166,514	486	2.7	7	33.9	34.8	55.0	580	0.242
253	110.344	1,176,110	490	2.9	7	34.0	35.3	54.6	582	0.243
254	137.547	1,430,404	596	2.5	6	34.6	35.8	67.8	705	0.294
255	106.218	1,089,746	454	2.4	6	36.9	37.9	52.1	535	0.223

Comparing the above with the corresponding results for population size 3,000 shows significant differences in effort in every case, but no significant difference in average effort. It seems that the amount effort to produce every result contains a lot of "noise", but the average effort changes slowly with parameter changes.