**Homework #6**

1. My homework 6 was implemented using Python 3 scripts from command line. Backtracking search was implemented recursively with MRV and degree heuristic. Variables were arbitrarily assigned 0 before 1 if the variable was selected and both values were possible. The experiments were conducted on a Windows machine with Intel i5-6600K (3.50GHz, 4 Cores) CPU. No existing code was implemented.
2. First 10 Assignments
   1. Example 1

|  |  |  |  |
| --- | --- | --- | --- |
|  | Variable Selected | Value Assigned | Number of Variables Affected by FC |
| Step 1 | 47 | 0 |  |
| Step 2 | 49 | 0 |  |
| Step 3 | 58 | 0 |  |
| Step 4 | 20 | 0 |  |
| Step 5 | 21 | 1 |  |
| Step 6 | 26 | 0 |  |
| Step 7 | 29 | 0 |  |
| Step 8 | 30 | 1 |  |
| Step 9 | 31 | 0 |  |
| Step 10 | 48 | 0 |  |

* 1. Example 2

|  |  |  |  |
| --- | --- | --- | --- |
|  | Variable Selected | Value Assigned | Number of Variables Affected by FC |
| Step 1 | 39 | 0 |  |
| Step 2 | 40 | 0 |  |
| Step 3 | 1 | 1 |  |
| Step 4 | 38 | 0 |  |
| Step 5 | 41 | 0 |  |
| Step 6 | 2 | 0 |  |
| Step 7 | 37 | 1 |  |
| Step 8 | 42 | 0 |  |
| Step 9 | 3 | 1 |  |
| Step 10 | 36 | 0 |  |

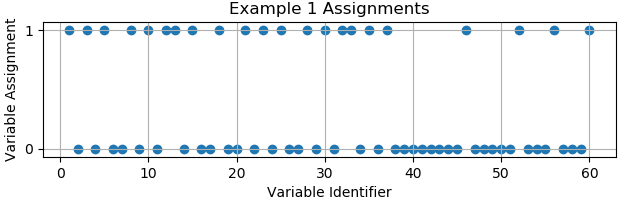
* 1. Example 3

|  |  |  |  |
| --- | --- | --- | --- |
|  | Variable Selected | Value Assigned | Number of Variables Affected by FC |
| Step 1 | 64 | 0 |  |
| Step 2 | 97 | 0 |  |
| Step 3 | 44 | 0 |  |
| Step 4 | 41 | 0 |  |
| Step 5 | 30 | 0 |  |
| Step 6 | 10 | 0 |  |
| Step 7 | 45 | 0 |  |
| Step 8 | 11 | 0 |  |
| Step 9 | 42 | 0 |  |
| Step 10 | 71 | 0 |  |

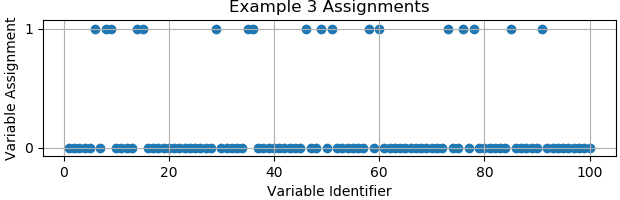
* 1. Example 4

|  |  |  |  |
| --- | --- | --- | --- |
|  | Variable Selected | Value Assigned | Number of Variables Affected by FC |
| Step 1 | 82 | 0 |  |
| Step 2 | 64 | 0 |  |
| Step 3 | 38 | 0 |  |
| Step 4 | 20 | 0 |  |
| Step 5 | 42 | 0 |  |
| Step 6 | 53 | 0 |  |
| Step 7 | 58 | 0 |  |
| Step 8 | 61 | 0 |  |
| Step 9 | 48 | 0 |  |
| Step 10 | 97 | 0 |  |

1. Final Results
   1. Example 1



* + 1. The solution plot is above. A full assignment of each point can be found in the assignment results output.
    2. Total number of assignments: **68**
    3. CPU execution time: **219.542ms** (0.219542 sec)
  1. Example 2
     1. Solution: **None found** (stopped after one hour)
     2. Total number of assignments: **3,629,523 (**in one hour)
     3. CPU execution time: **N/A** (~one hour)
  2. Example 3



* + 1. The solution plot is above. A full assignment of each point can be found in the assignment results output.
    2. Total number of assignments: **101**
    3. CPU execution time: **549.068ms** (0.549068 sec)
  1. Example 4
     1. Solution: **None found** (stopped after one hour)
     2. Total number of assignments: **2,792,406 (**in one hour)
     3. CPU execution time: **N/A** (~one hour)