# Week 11 – In-Class Assignment

## Pure Backtracking

*Assume we use pure backtracking to search for a solution to this problem. We use a fixed variable ordering in the search (V1, V2, V3, V4) and the values are considered in the order shown in the picture above. Then, show the order in which individual variable assignments are considered by backtracking (this is analogous to the order in which nodes are expanded by depth-first search).*

Iterations:

1. 1R
2. 1R, 2B
3. 1R, 2B, 3B (4X)
4. 1R, 2B, 3G (4X)
5. 1R, 2G
6. 1R, 2G, 3B (4X)
7. 1R, 2G, 3G
8. 1R, 2G, 3G, 4B (Solved!)
9. 1B
10. 1B, 2G

## Backtracking with Forward Checking

*Repeat assuming we use backtracking with forward checking to search for a solution to this problem. We use the same variable ordering and value ordering as before. Show the order in which assignments are considered by BT-FC. Whenever propagating after an assignment causes a domain to become empty, that causes backup in the search. Assume that the search continues even after finding a valid solution. Write each assignment (up to 10) as before.*

1. 1R
2. 1R, 2B (4X)
3. 1R, 2G
4. 1R, 2G, 3B (4X)
5. 1R, 2G, 3G, 4B (Solved!)
6. 1B
7. 1B, 2G
8. 1B, 2G, 3G
9. 1B, 2G, 23G, 4B (Solved!)
10. 1G

## Arc Consistency

*Write the value in each of the indicated value domains after any changes required to achieve arc consistency for just that arc. Then, assume that the following arcs are done sequentially, with the effects on the domains propagating. Write domains as a sequence of letters ( R B ). If there are no other values in the domain, write None.*

V1 – V2 : D1 = RGB; D2 = GB

V1 – V3 : D1 = RGB; D3 = GB

V2 – V4 : D2 = G ; D4 = B

V3 – V4 : D3 = G; D4 = B

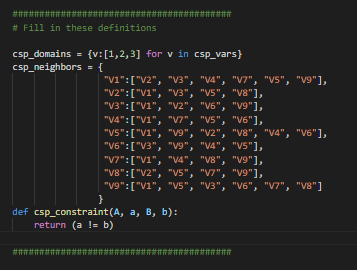
V1 – V2 : D1 = RB; D2 = G

V1 – V3 : D1 = RB; D3 = G

## Thinking of CSP

|  |  |
| --- | --- |
|  | **True** |
| **False** |
| **True** |
| **True** |
| **False** |
| **False** |
|  | **True** |
| **True** |
| **True** |
| **False** |
| **True** |

## Semi-Magic Square



Result:

