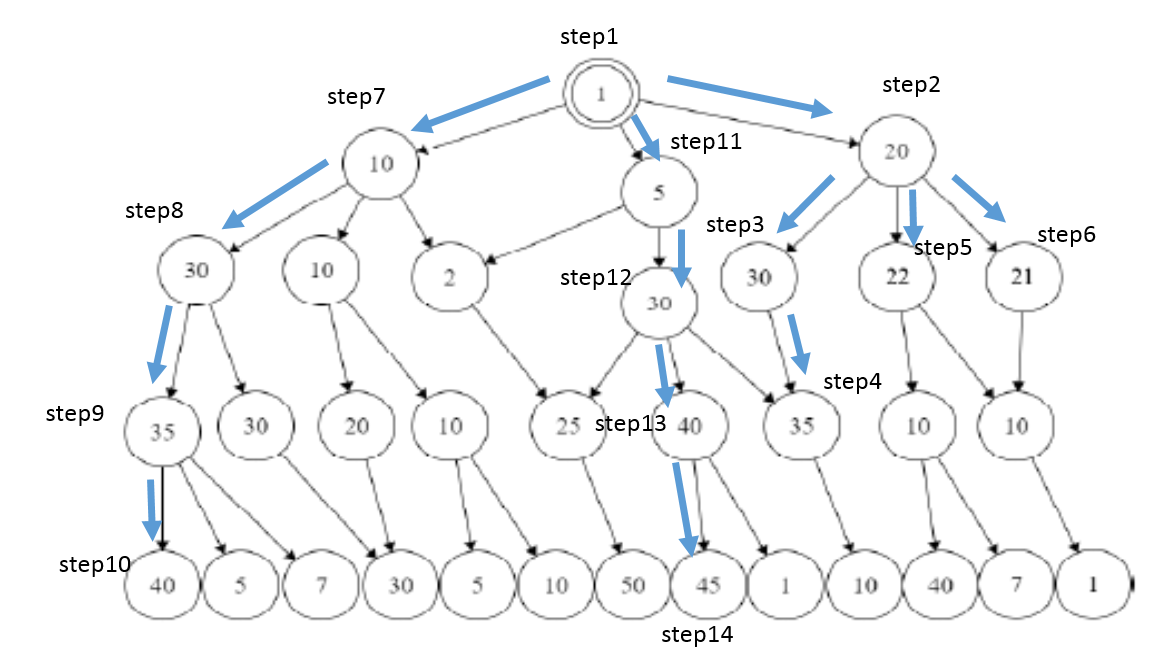
**Assignment2**

**Name: Kaichen Zhang ID:40000160**

**Question1:**

**Answer:**

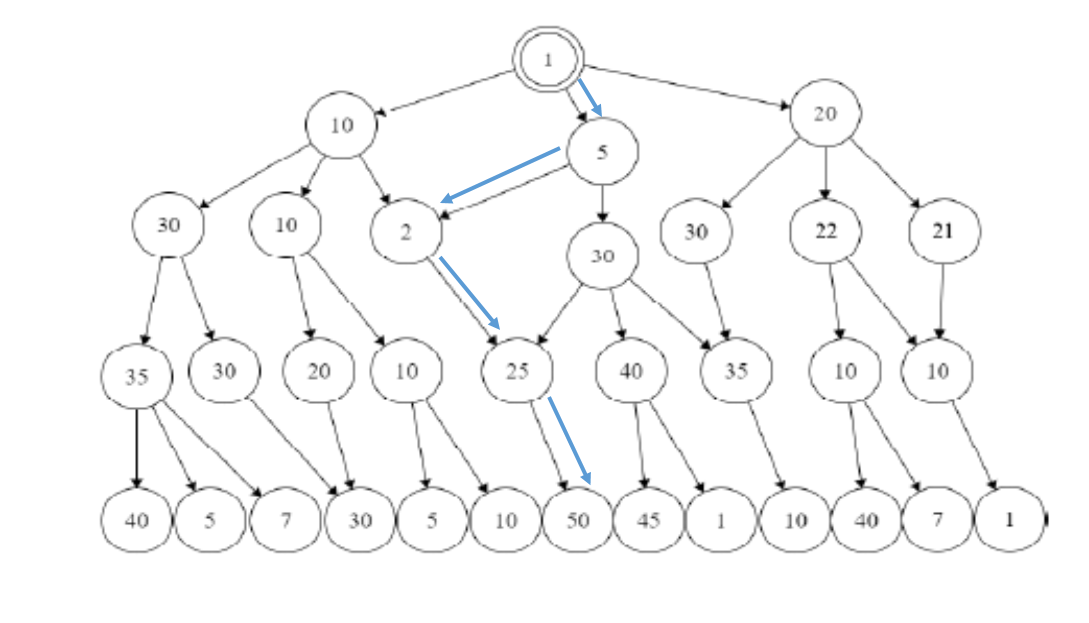
**Hill Climbing Search (with stacks to backtracking and assume the algorithm handles the loop):**



|  |  |  |  |
| --- | --- | --- | --- |
| **Steps** | **Visited Node** | **Successors** | **Sorts and Prune** |
| **1** | **1** | **10, 5, 20** | **20, 10, 5** |
| **2** | **20** | **30, 22, 21** | **30, 22, 21** |
| **3** | **30** | **35** | **35** |
| **4** | **35** | **10** |  |
| **5** | **22** | **10, 10** |  |
| **6** | **21** | **10** |  |
| **7** | **10** | **30, 10, 2** | **30** |
| **8** | **30** | **35, 30** | **35** |
| **9** | **35** | **40, 5, 7** | **40** |
| **10** | **40** |  |  |
| **11** | **5** | **2, 30** | **30** |
| **12** | **30** | **25, 40, 35** | **40** |
| **13** | **40** | **45, 1** | **45** |
| **14** | **45** |  |  |

**The output of the algorithm is 45.**

**Simulated Annealing Search:**



**From the start node 1, we have three successors, by using random value, we choose the second neighbor which is 5. Then from two successors of 5, we choose the first child by using the second random value 0.25. Because P(5,2,T)=1/2>0.25, we can accept the new value. Then the only successor of node 2 is bigger than itself, we traverse the node. The same to node 50, and reaching the end condition.**

|  |  |  |
| --- | --- | --- |
| **Steps** | **Current Node** | **Visited** |
|  | **1** |  |
|  | **5** | **1** |
|  | **2** | **1,5** |
|  | **25** | **1,5,2** |
|  | **50** | **1,5,2,25** |
|  |  | **1,5,2,25,50** |

**Output is 50.**

**Question2:**

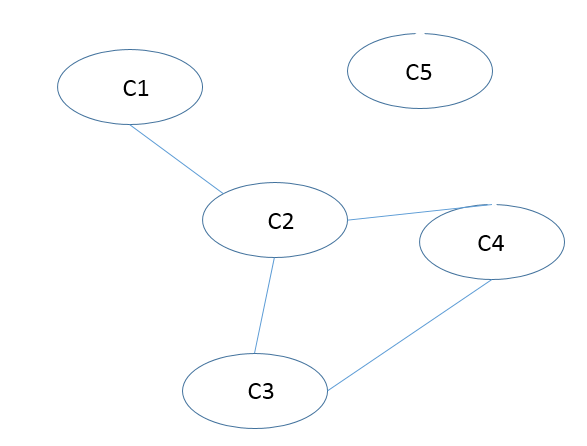
**Answer:**

**a)**

|  |  |
| --- | --- |
| **Variables** | **Domains** |
| **C1** | **{A,C}** |
| **C2** | **{A}** |
| **C3** | **{B,C}** |
| **C4** | **{B,C}** |
| **C5** | **{A,B}** |

**Constrains: according to the time slot, we have C1≠C2, C2≠C3, C2≠C4, C3≠C4.**

**b) The constraint graph**



**c)**

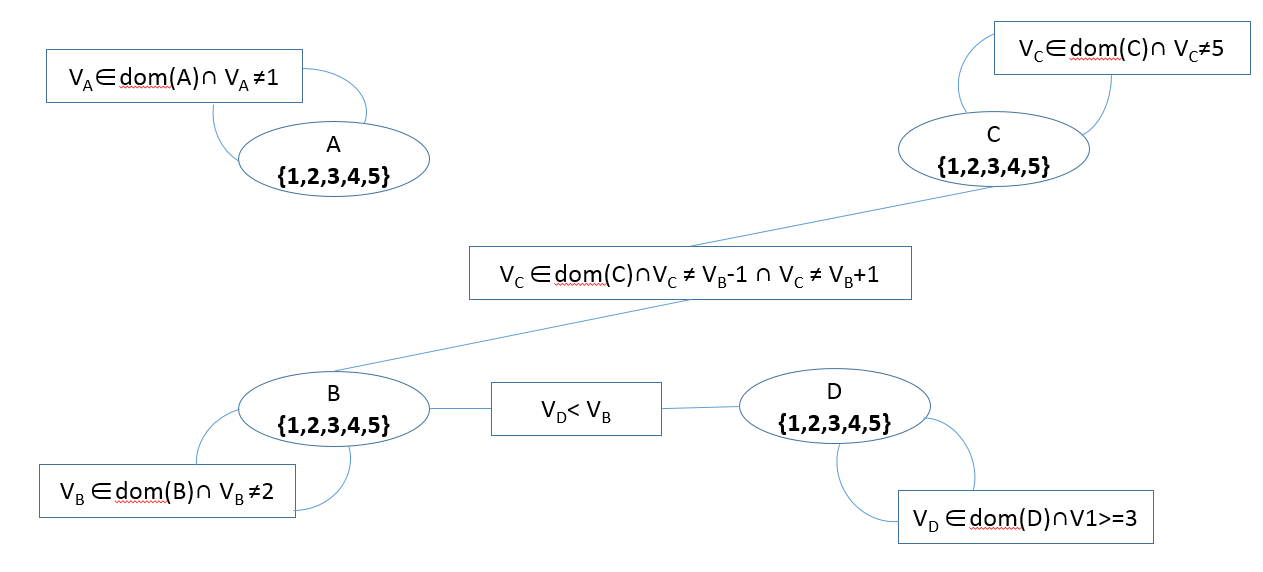
**We prefer to solve tree structured because it’s efficient, it can be solved within polynomial time. For a general CSP graph search it may take an exponential time in the worst case.**

**Question3:**

**Answer:**

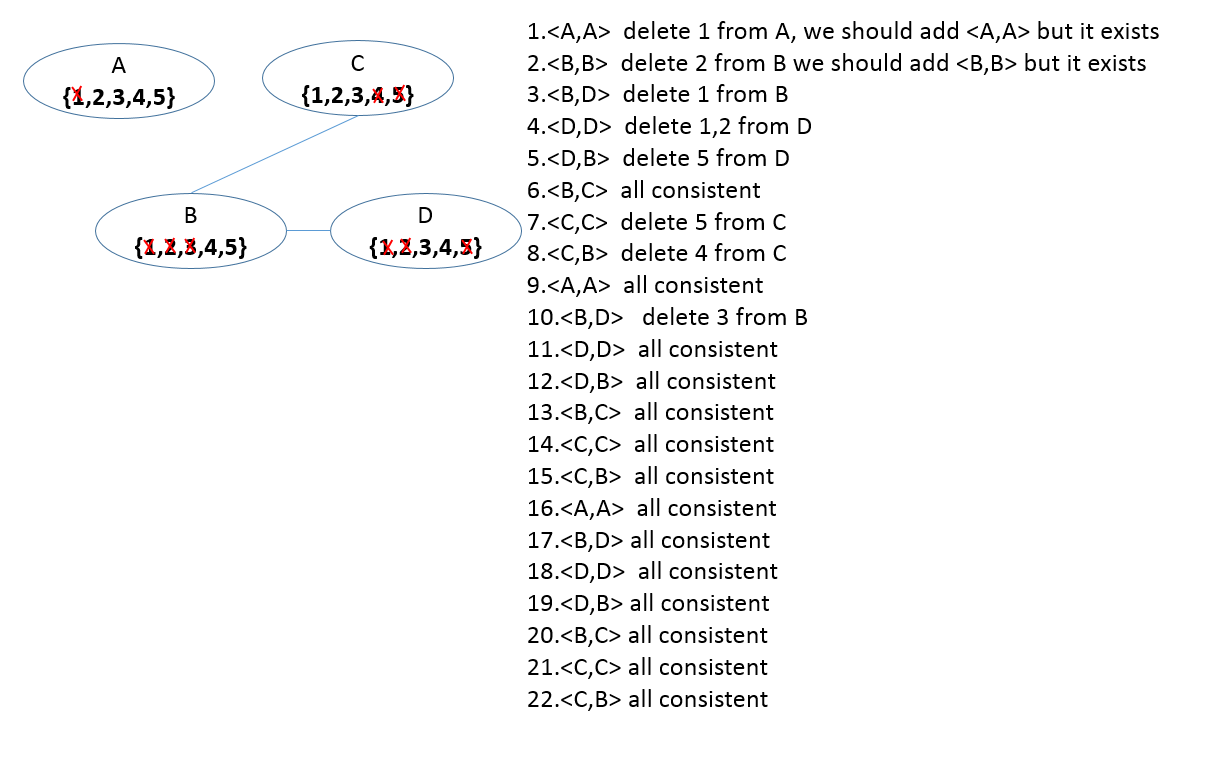
**a)**

**The constraint network is:**



**So we have arcs {<A,A>,<B,B>,<D,D>,<D,B>,<C,B>,<C,C>}**

**Implement DFS on the constraint network:**



**With heuristic, check the next node’s successive arc first.**