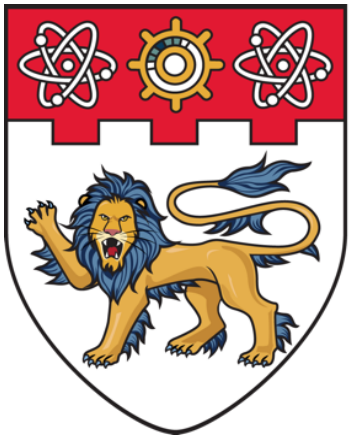


NANYANG TECHNOLOGICAL UNIVERSITY
SINGAPORE

CZ3005: Artificial Intelligence

Problem Solving Lab Assessment

Shantanu Sharma

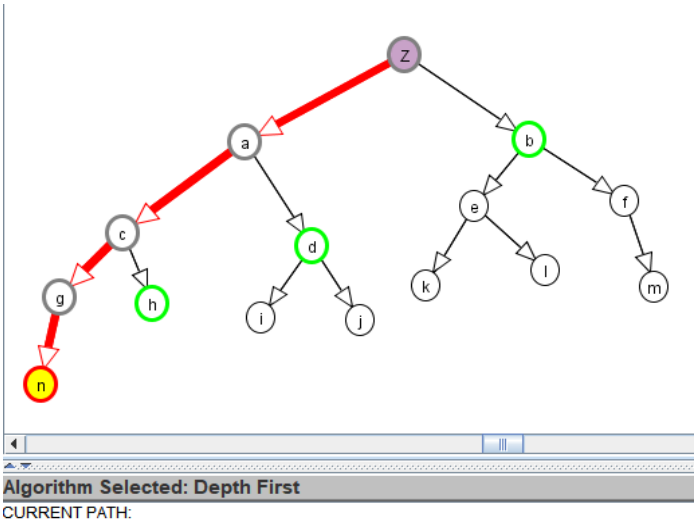
U1622895F

Lab Group: SSP3

1. Question One (ordered from left to right):

a) Assume arc cost to be **one**, start node as "z" and goal node as "n":

Depth First



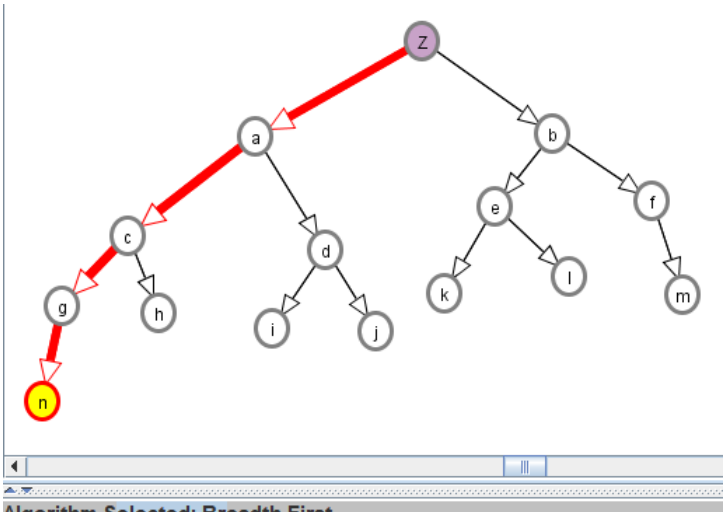
Algorithm Selected: Depth First
CURRENT PATH:
Z --> a --> c --> g --> n (Goal)
Path to last Goal Node: Z --> a --> c --> g --> n (Goal) Cost: 4.0
Nodes expanded: 5

NEW FRONTIER:

Node: h	Path Cost: 3.0	Path: Z --> a --> c --> h
Node: d	Path Cost: 2.0	Path: Z --> a --> d
Node: b	Path Cost: 1.0	Path: Z --> b

In the above graph DFS expands only 5 nodes

Breadth First



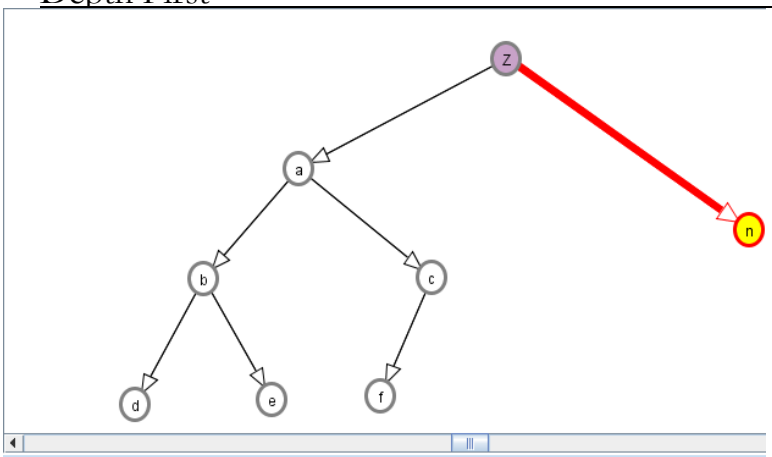
Algorithm Selected: Breadth First
CURRENT PATH:
Z --> a --> c --> g --> n (Goal)
Path to last Goal Node: Z --> a --> c --> g --> n (Goal) Cost: 4.0
Nodes expanded: 15

NEW FRONTIER:
No node on frontier.

In the given graph BFS expands every node (total 15)

b) Assume arc cost to be **one**, start node as "z" and goal node as "n":

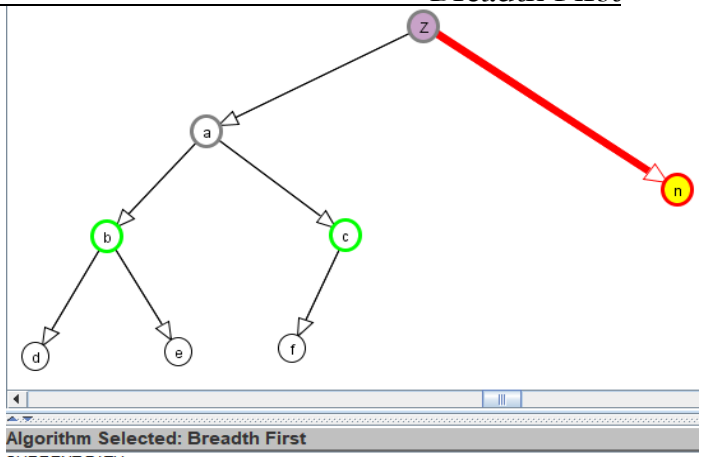
Depth First



Algorithm Selected: Depth First
CURRENT PATH:
Z --> n (Goal)
Path to last Goal Node: Z --> n (Goal) Cost: 1.0
Nodes expanded: 8

In the above graph DFS expands every node (z>a>b>d>e>d>f>n)

Breadth First



Algorithm Selected: Breadth First
CURRENT PATH:
Z --> n (Goal)
Path to last Goal Node: Z --> n (Goal) Cost: 1.0
Nodes expanded: 3

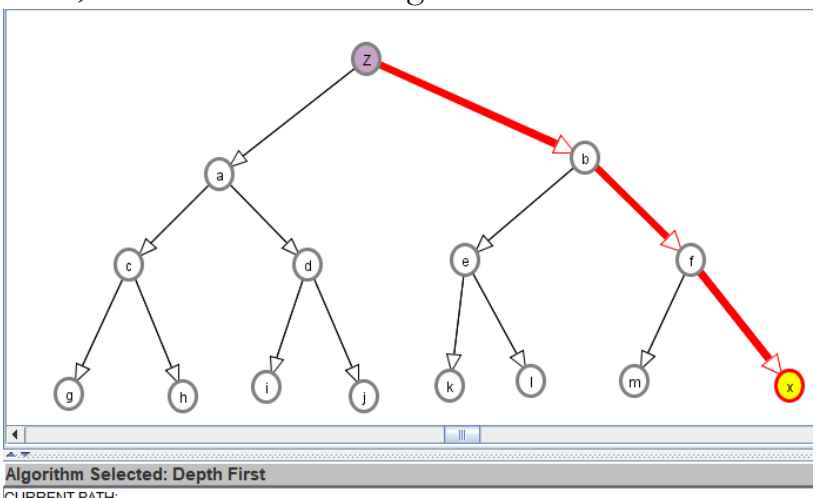
NEW FRONTIER:

Node: b	Path Cost: 2.0	Path: Z --> a --> b
Node: c	Path Cost: 2.0	Path: Z --> a --> c

In the given graph BFS expands only 3 nodes

c) Assume arc cost to be **one**, start node as "z" and goal node as "x":

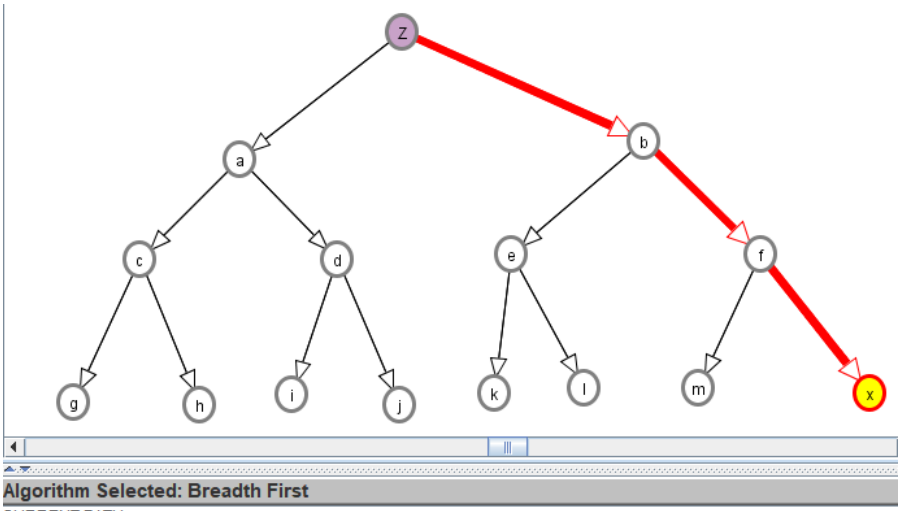
Depth first search:
All nodes expanded



Algorithm Selected: Depth First
CURRENT PATH:
Z --> b --> f --> x (Goal)
Path to last Goal Node: Z --> b --> f --> x (Goal) Cost: 3.0
Nodes expanded: 15

NEW FRONTIER:
No node on frontier.

Breadth first search:
All nodes expanded

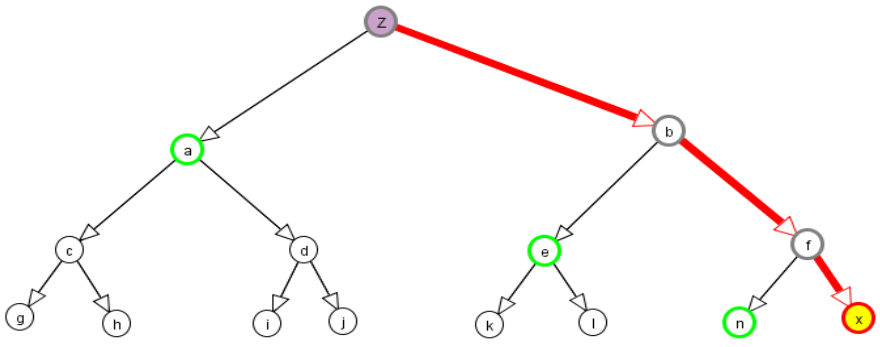


Algorithm Selected: Breadth First
CURRENT PATH:
Z --> b --> f --> x (Goal)
Path to last Goal Node: Z --> b --> f --> x (Goal) Cost: 3.0
Nodes expanded: 15

NEW FRONTIER:
No node on frontier.

Paths added to frontier:
Z --> b --> f --> n, Z --> b --> f --> x (Goal)

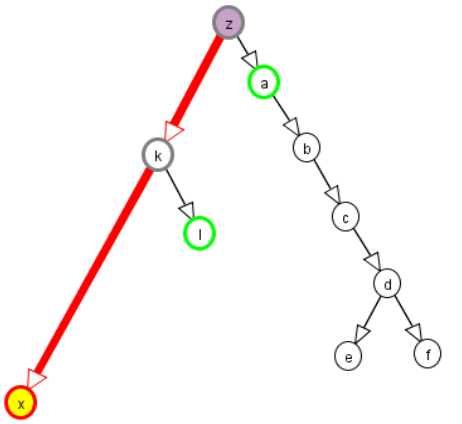
A* search:
4 nodes expanded



Algorithm Selected: A*
CURRENT PATH:
Z --> b --> f --> x (Goal)
Path to last Goal Node: Z --> b --> f --> x (Goal) Cost: 3.0
Nodes expanded: 4

NEW FRONTIER:

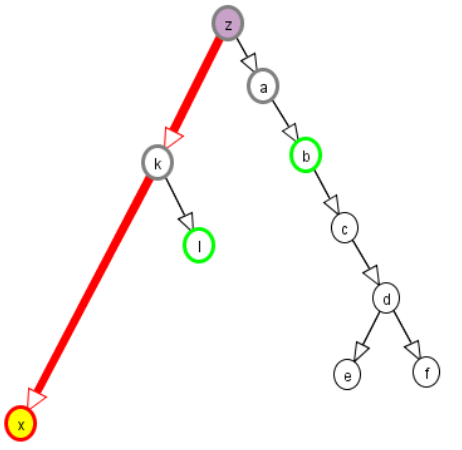
d) Assume **Euclidean distance** as the arc cost and as the heuristic function, start node as "z" and goal node as "x":



Algorithm Selected: Depth First
CURRENT PATH:
z --> k --> x (Goal)
Path to last Goal Node: z --> k --> x (Goal) Cost: 7.0
Nodes expanded: 3

NEW FRONTIER:
Node: l Path Cost: 3.0 Path: z --> k --> l
Node: a Path Cost: 1.0 Path: z --> a

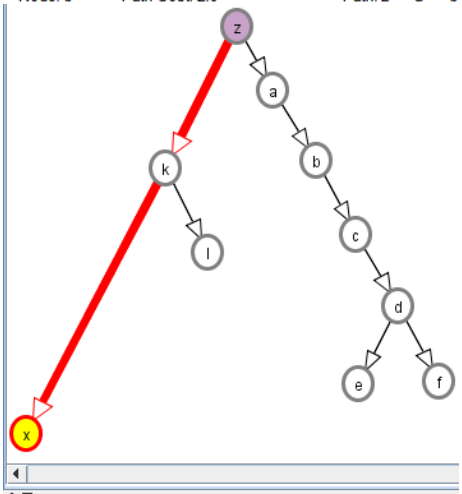
Depth First Search:
Nodes Traversed= 3



Algorithm Selected: Breadth First
CURRENT PATH:
z --> k --> x (Goal)
Path to last Goal Node: z --> k --> x (Goal) Cost: 7.0
Nodes expanded: 4

NEW FRONTIER:
Node: l Path Cost: 3.0 Path: z --> k --> l
Node: b Path Cost: 2.0 Path: z --> a --> b

Breadth First Search:
Nodes Traversed= 4

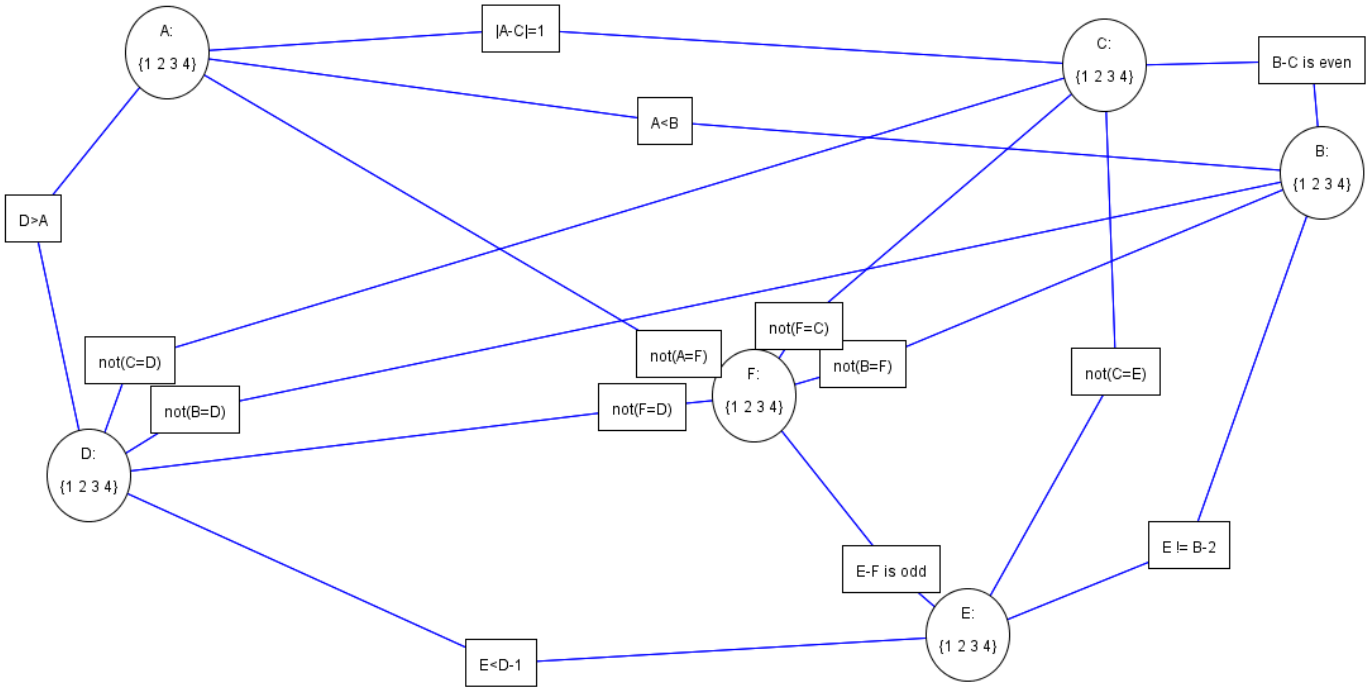


Algorithm Selected: A*
CURRENT PATH:
z --> k --> x (Goal)
Path to last Goal Node: z --> k --> x (Goal) Cost: 7.0
Nodes expanded: 10

Breadth First Search:
Nodes Traversed= 4

2. Question Two:

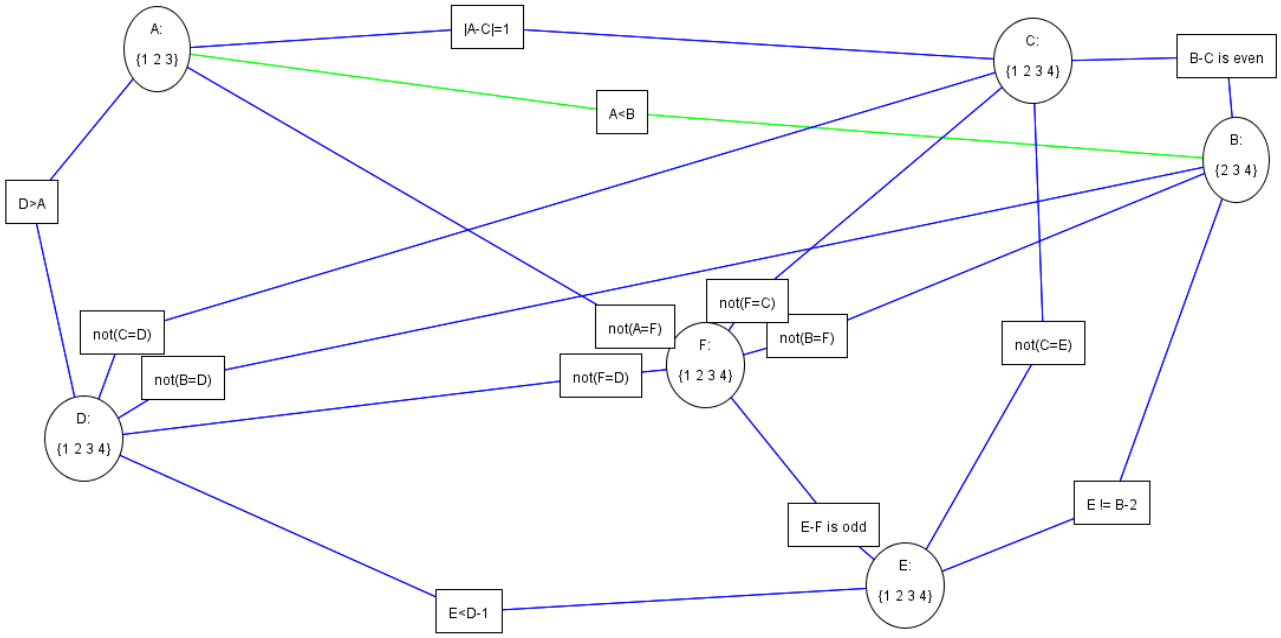
a) CSP graph:



b) First five steps:

STEP 1: Arc: <B,A> removes 1 from the domain of B

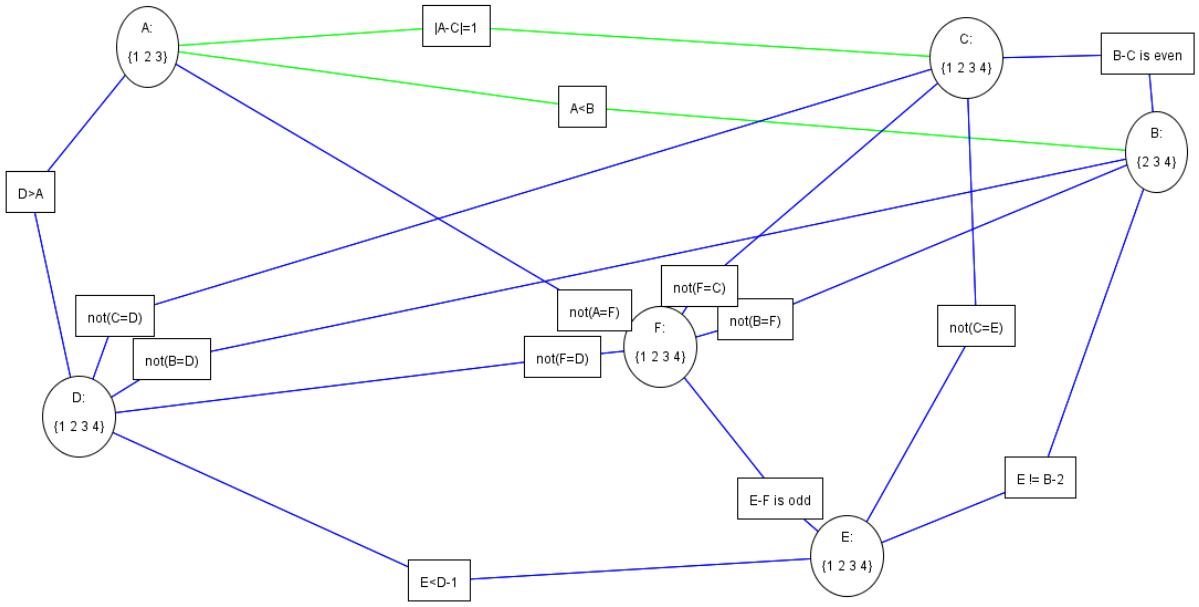
Arc: <A,B> removes 4 from the domain of A



STEP 2:

Arc: <C,A> is consistent

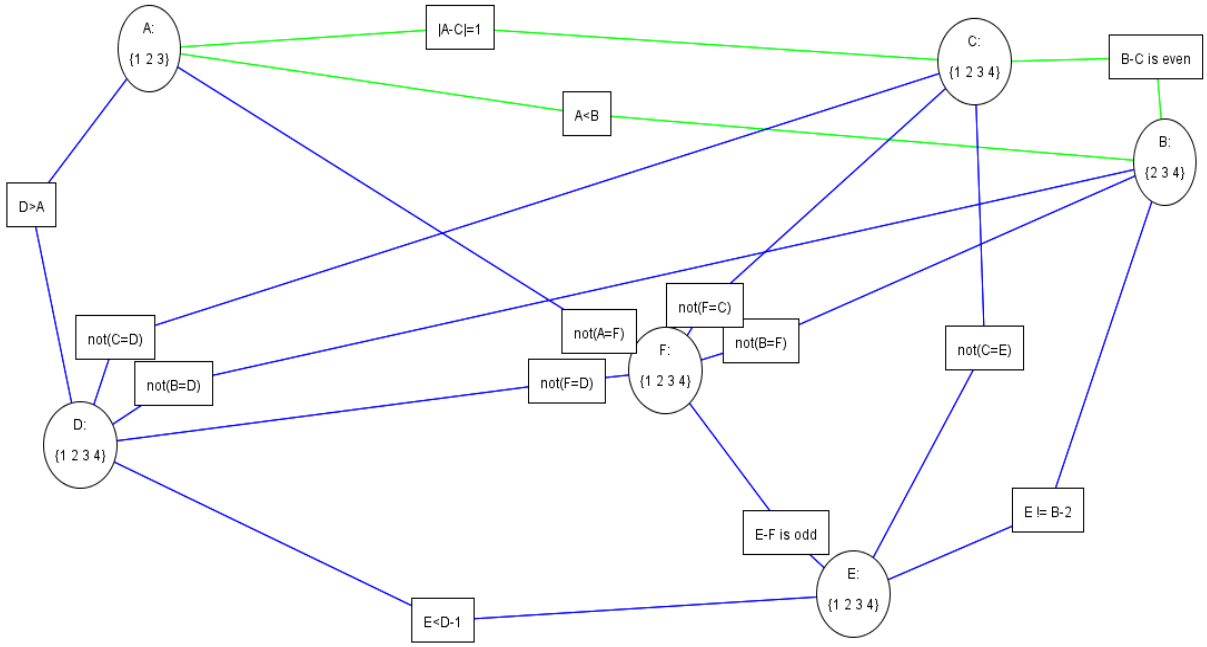
Arc: <A,C> is consistent



STEP 3:

Arc: <C,B> is consistent

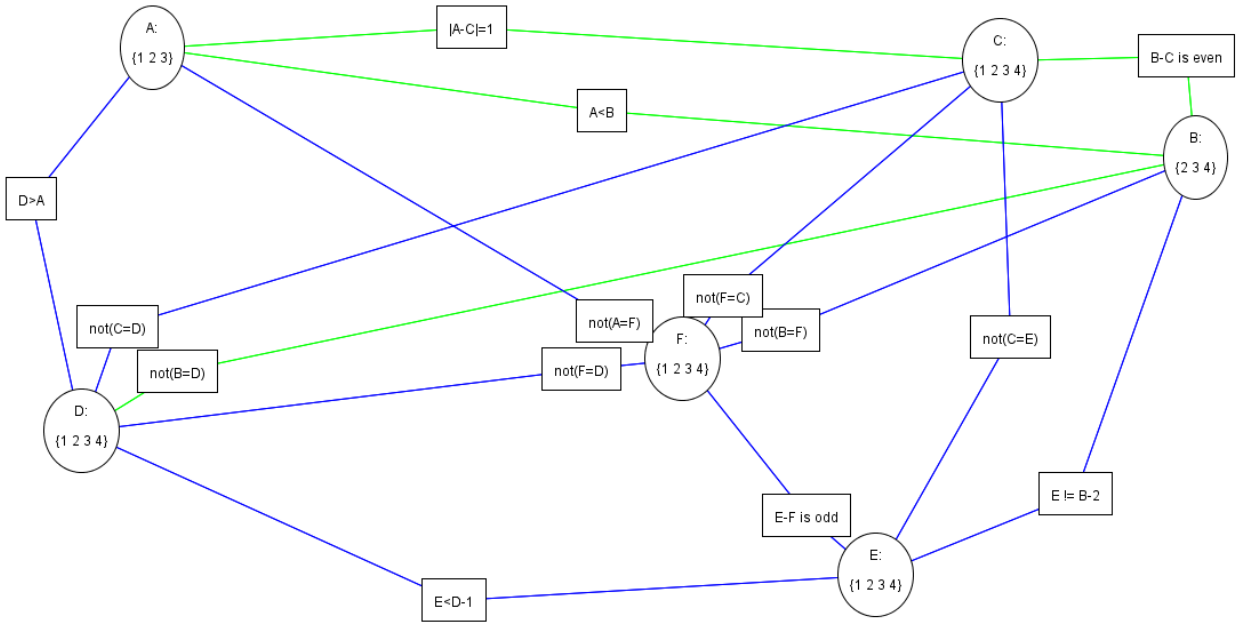
Arc: <B,C> is consistent



STEP 4:

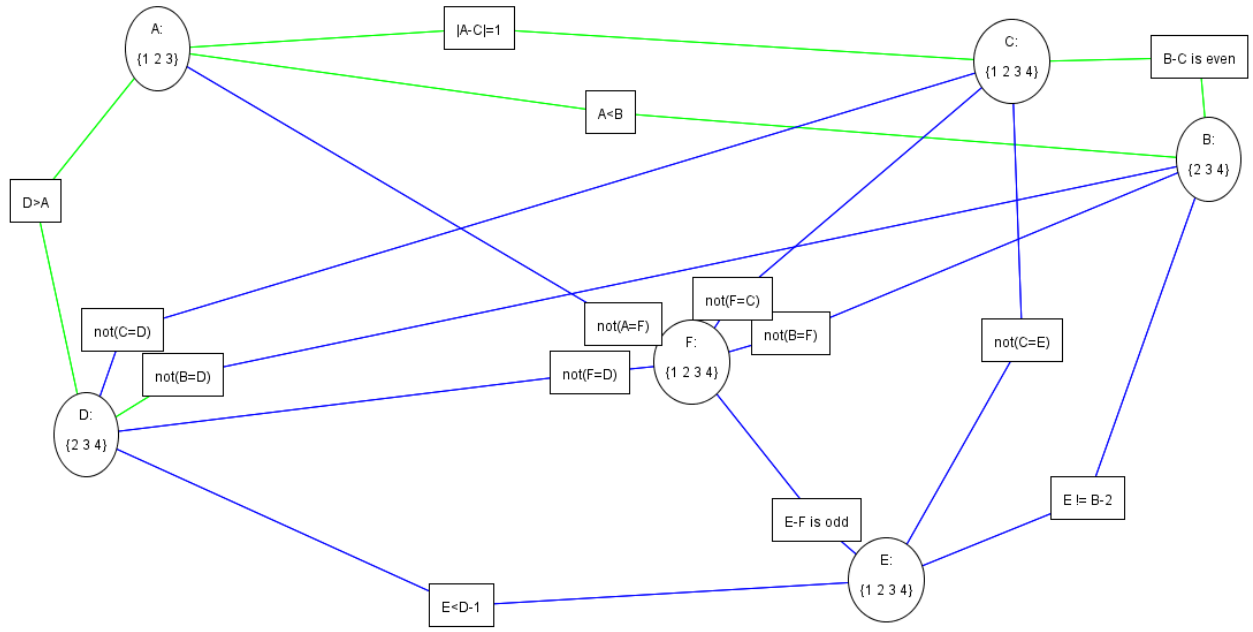
Arc: <D,B> is consistent

Arc: <B,D> is consistent

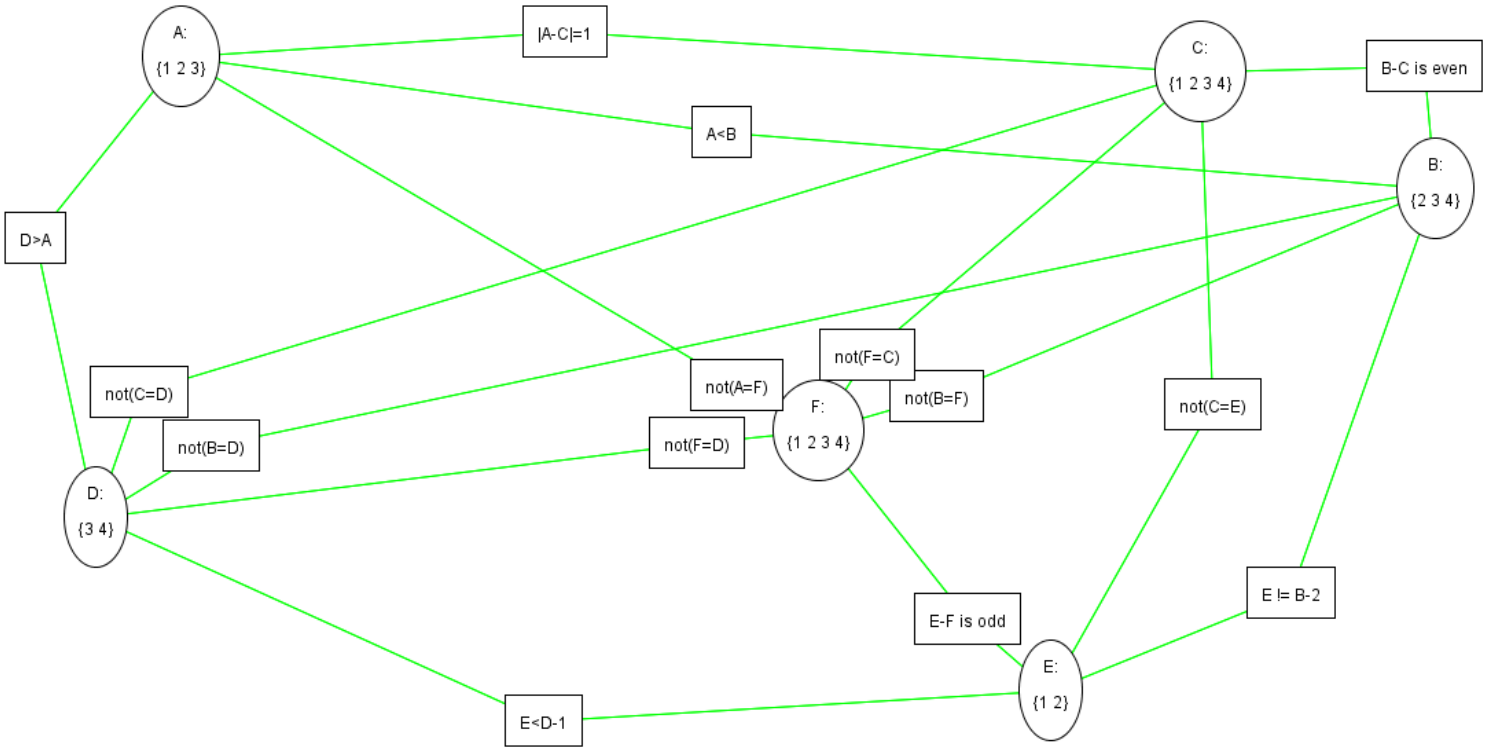


STEP 5: Arc: <A,D> is consistent

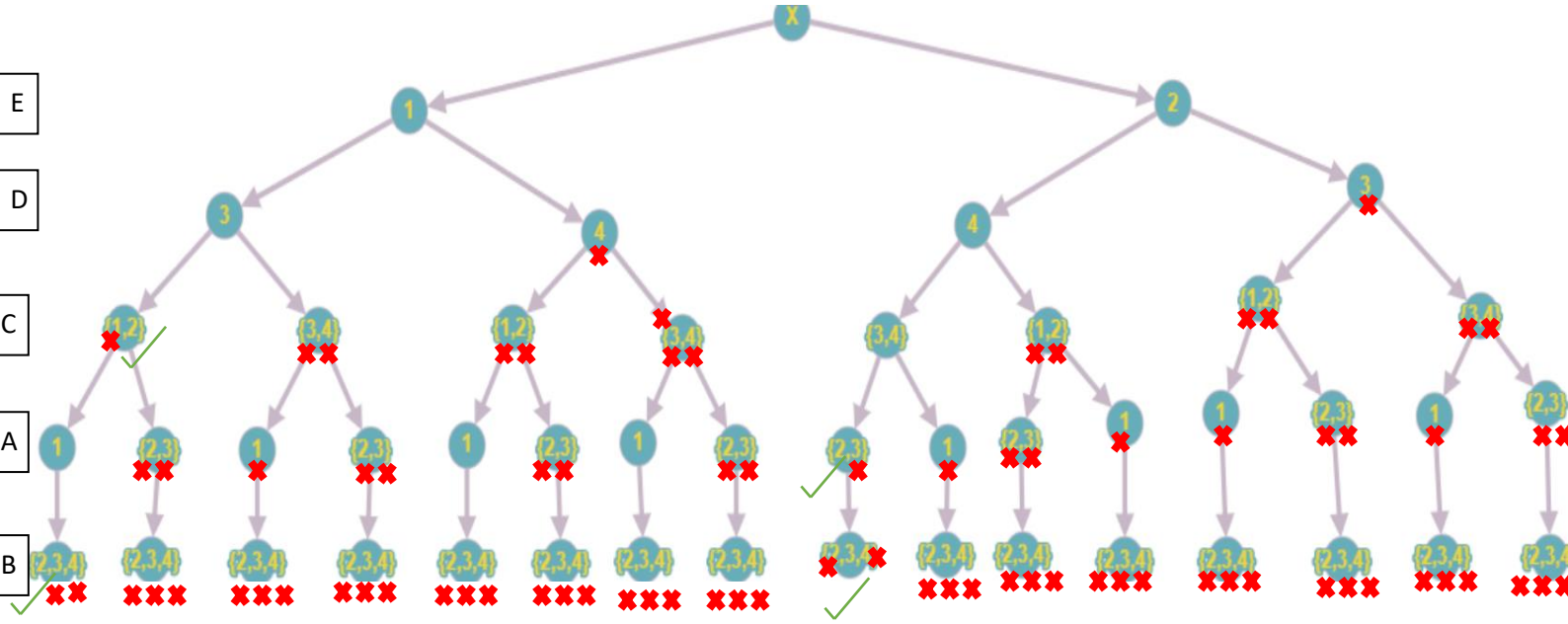
Arc: <D,A> removes 1 from the domain of D



c) Constraint Graph after consistency has stopped:



c) Tree of splits and final solution:




```
DOMAIN-SPLITTING HISTORY:

E in {1}
  D in {3}
    C in {1 2}
      A in {1}
        B in {2}
          Solution found:  A = 1, B = 2, C = 2, D = 3, E = 1, F = 4
        B in {3 4}
          Cannot split variable A
      A in {2 3}
        Cannot split variable A
    C in {3 4}
      Cannot split variable A
  D in {4}
    Cannot split variable A
E in {2}
  Solution found:  A = 2, B = 3, C = 3, D = 4, E = 2, F = 1
```

Solutions:

A=1, B=2, C=2, D=3, E=1, F=4

A=2, B=3, C=3, D=4, E=2, F=1

3. Question Three:

Effect of reducing *h(n)* when *h(n)* is already an underestimate:

More nodes are expanded, however for the case that same number of nodes are expanded, at least that many nodes are expanded. This statement might fail for some cases reason being that a different optimal path may be found. But we can conclude that for any sub-optimal path explored with h_1 will be explored with h_2 if $h_2(x) <= h_1(x) <= cost(x,y)$, where $cost(x,y)$ is the actual cost from x to y.

Performance A* when *h(n)* is the exact distance from *n* to a goal:

The results depend on existence of multiple optimal. If the frontier acts as a stack for nodes with equal *f*-values, then it will proceed to the goal without expanding any node off a single optimal path.

When *h(n)* is not an underestimate:

If $h(n) <= cost(n,g) + \epsilon$ (where $\epsilon >= 0$), the first path found will be at most ϵ from optimal.

If $h(n) <= cost(n,g) \times \gamma$ (where $\gamma >= 1$), the first path found will be at most γ times optimal.

[Reference: http://www.aispace.org/]

However, finding an optimal path may not be true always.

4. Question Four:

Approximately 4 and a half hours.