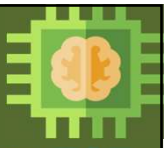
Elective Course

Course Code: CS4103

Autumn 2025-26



Lecture #15

Artificial Intelligence for Data Science

Week-4: PROBLEM SOLVING BY SEARCH

Adversarial Search Problem--- Games [Practice Example]

CONSTRAINT SATISFACTION PROBLEM (CSP)

Course Instructor:

Dr. Monidipa Das

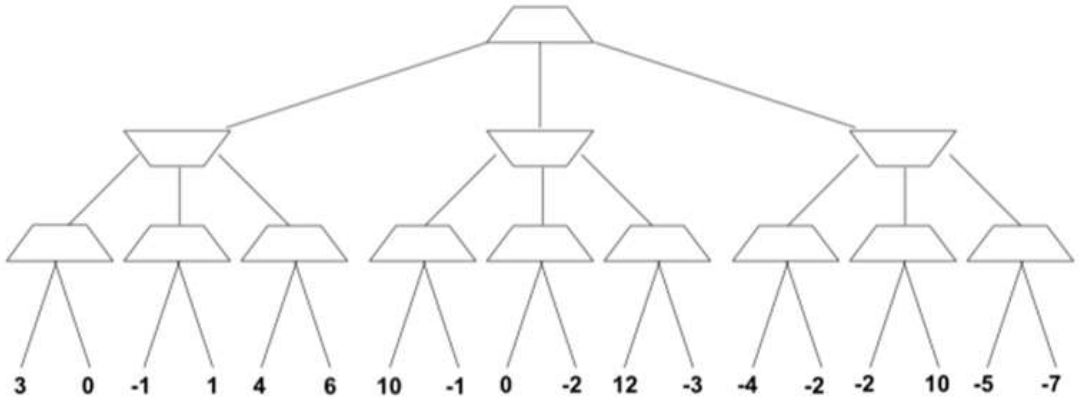
Assistant Professor

Department of Computational and Data Sciences

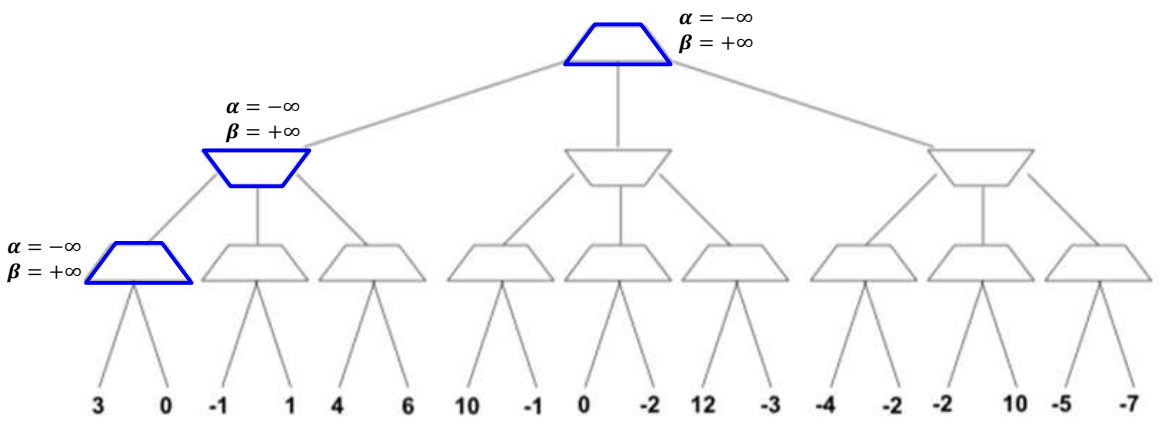
Indian Institute of Science Education and Research Kolkata, India 741246

Practice Problem

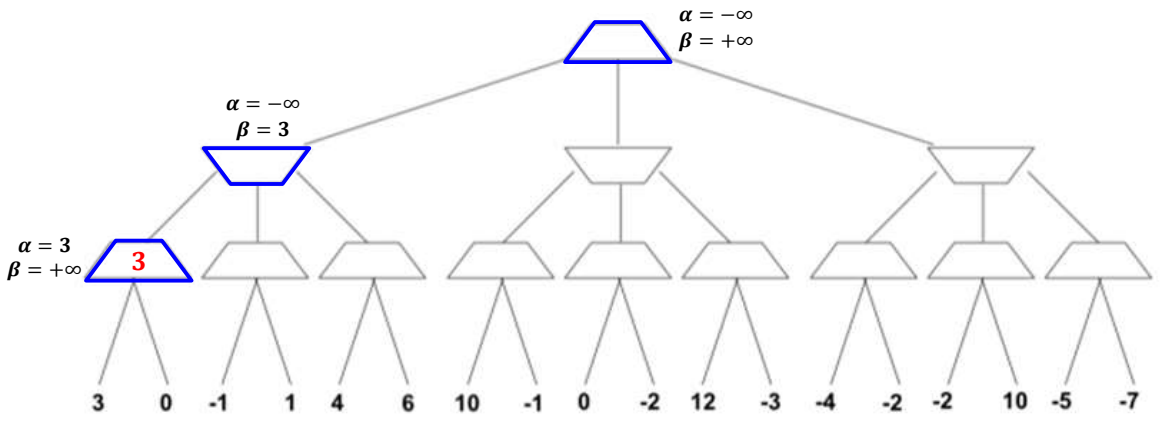
Which leaf nodes are never visited due to pruning?



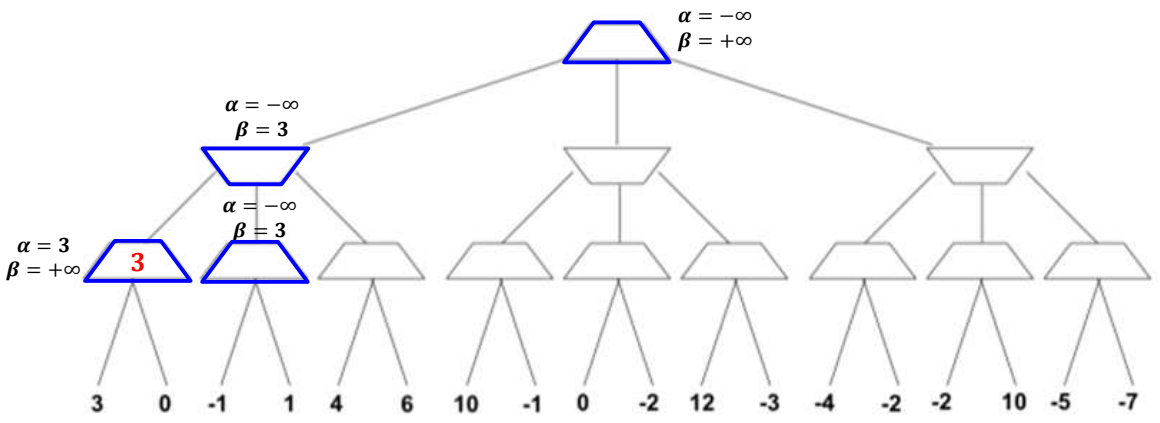
Solution



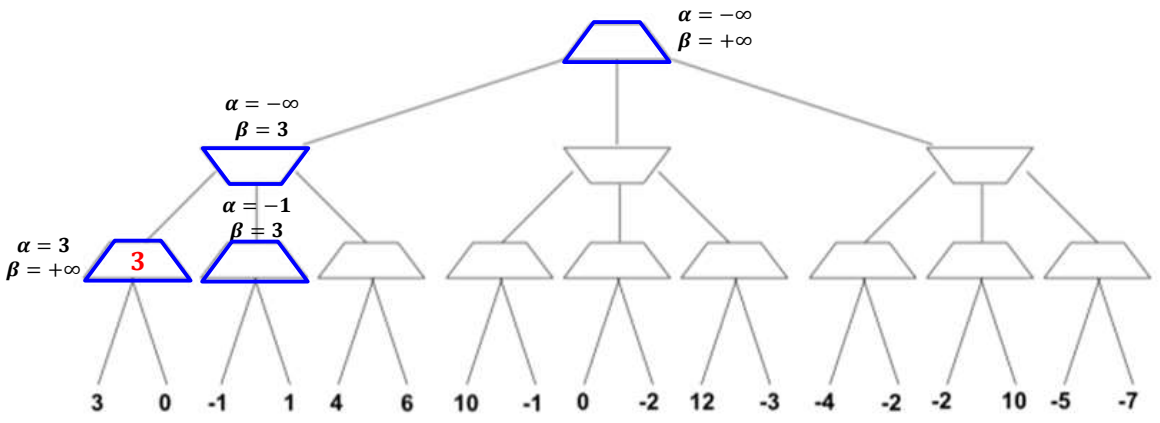
Solution



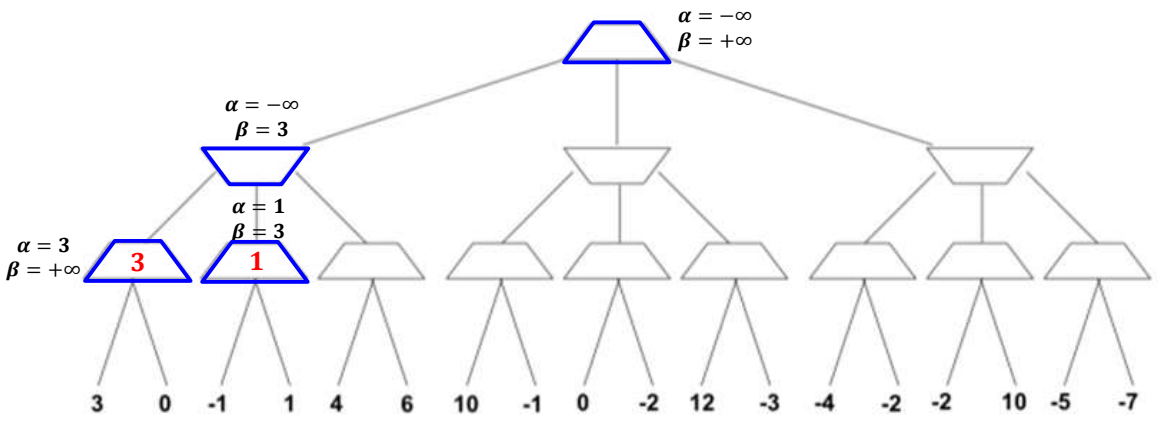
Solution



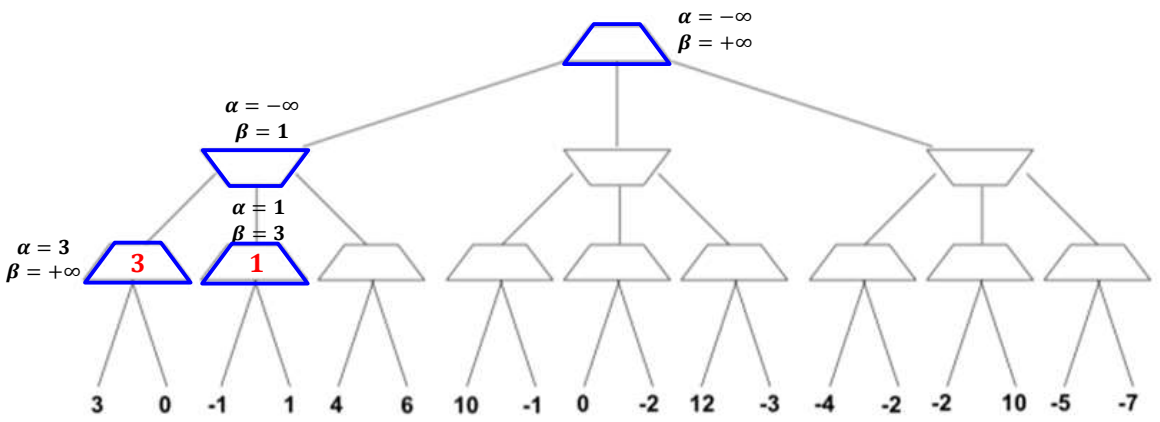
Solution



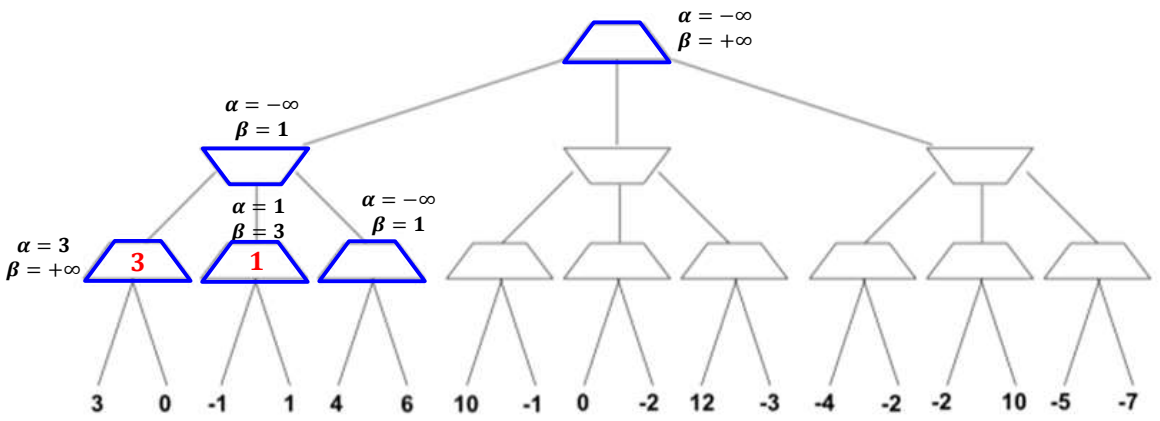
Solution



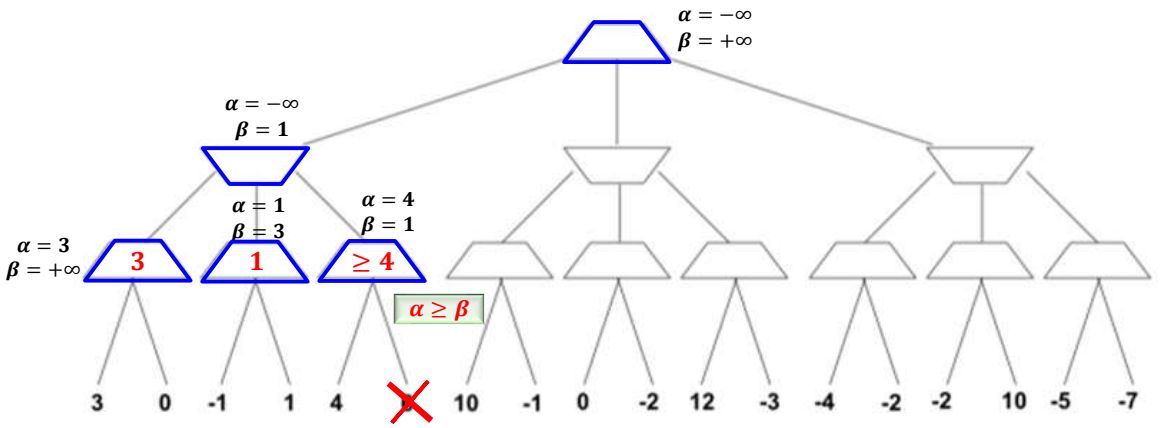
Solution



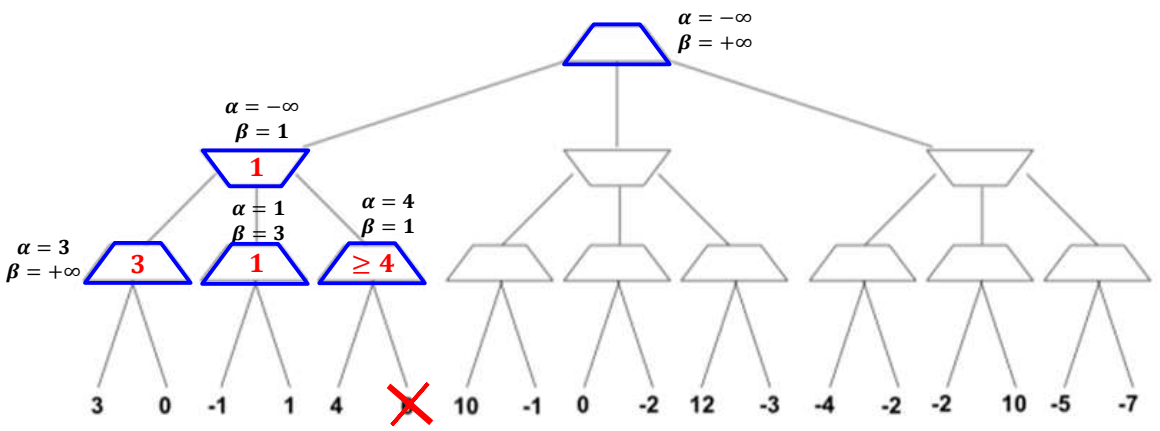
Solution



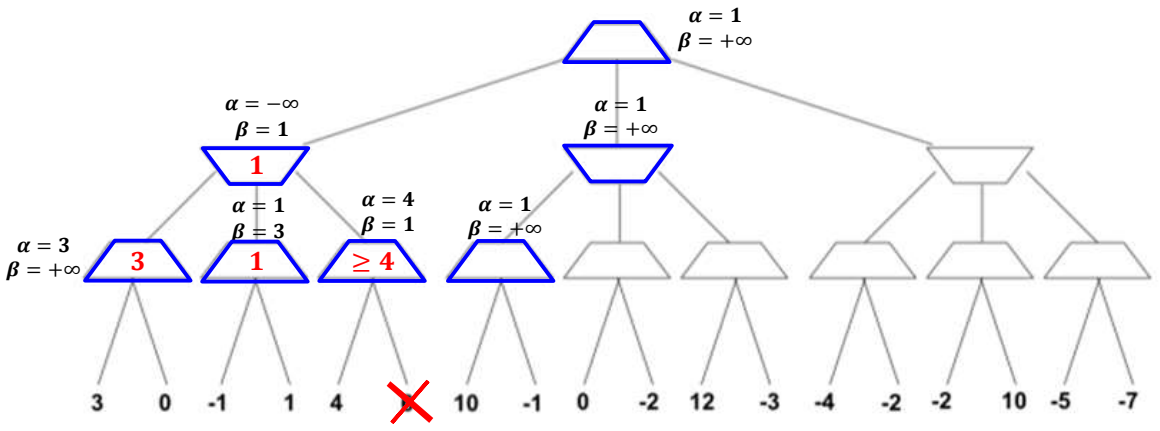
Solution



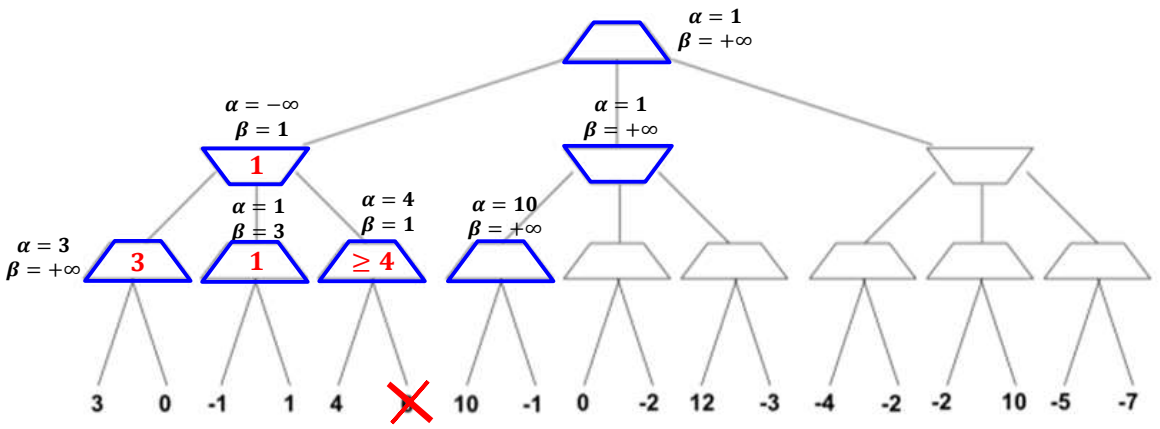
Solution



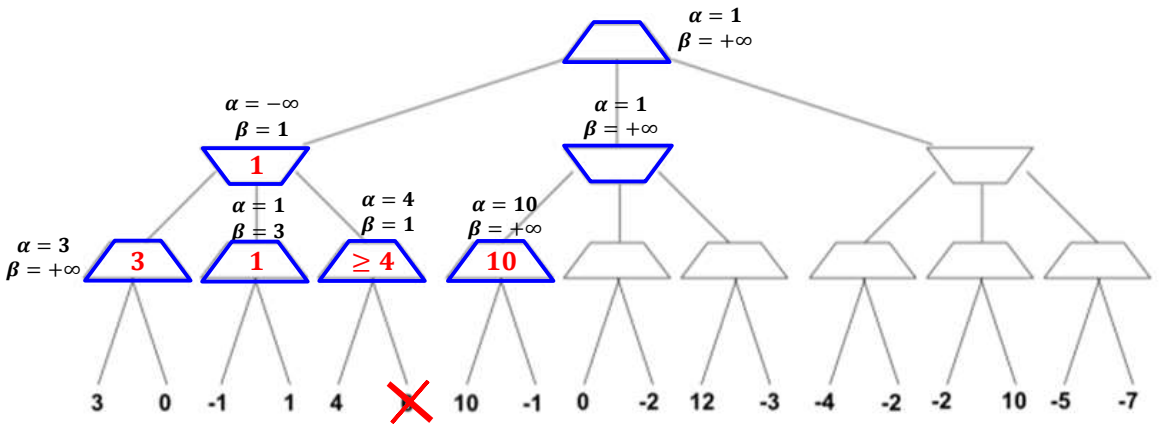
Solution



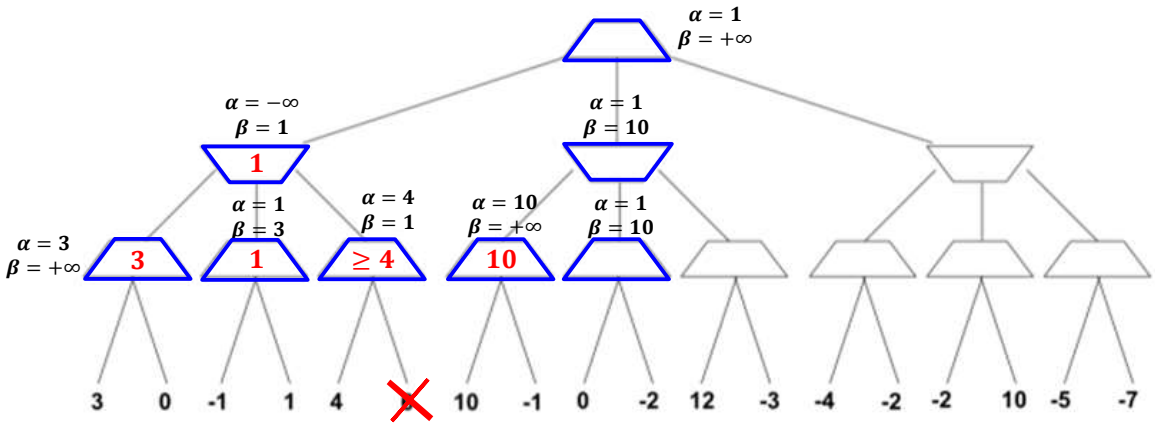
Solution



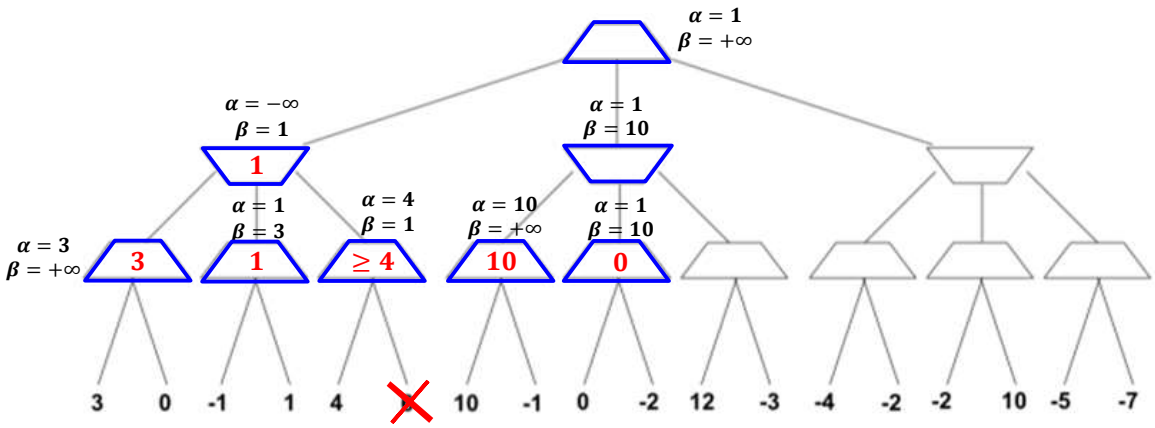
Solution



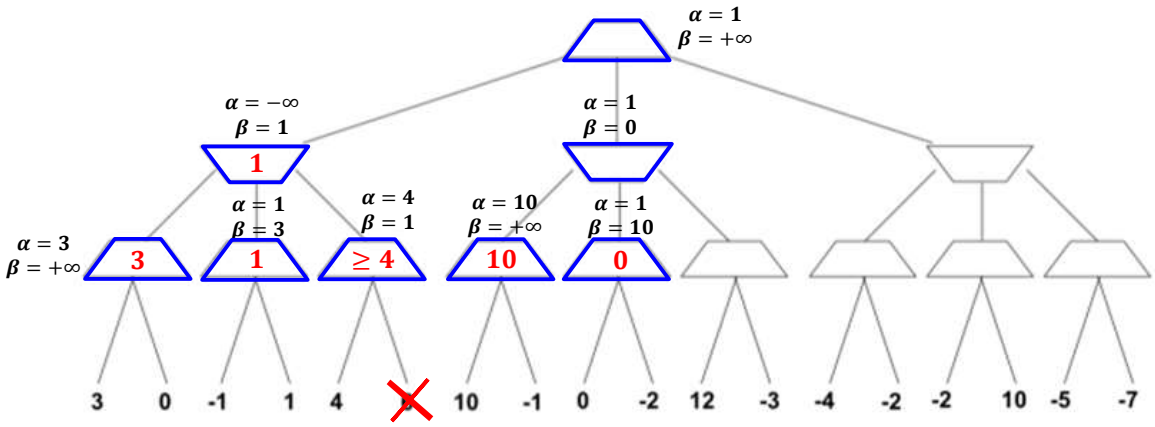
Solution



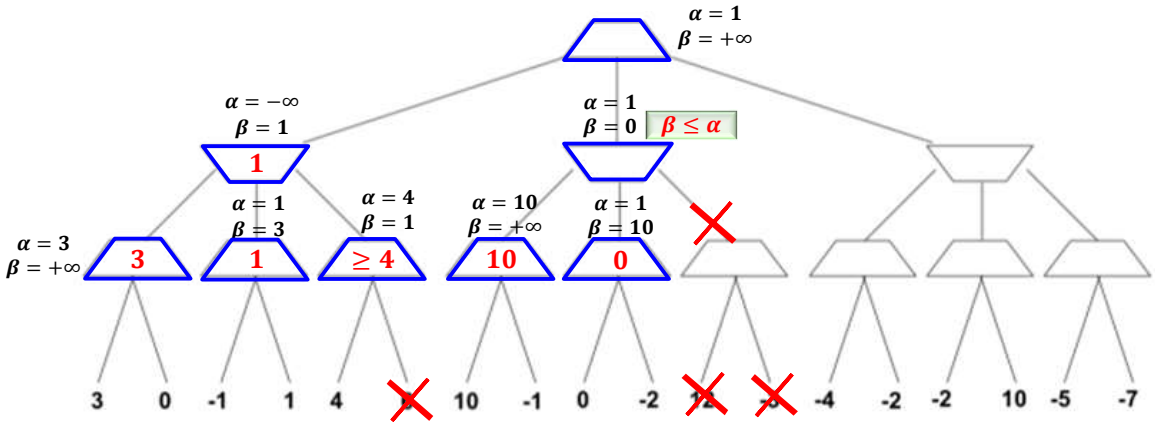
Solution



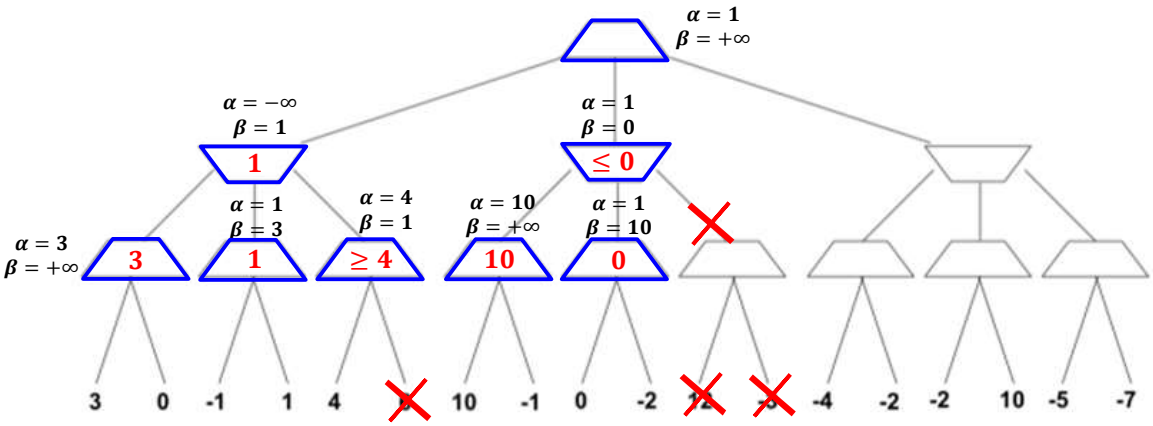
Solution



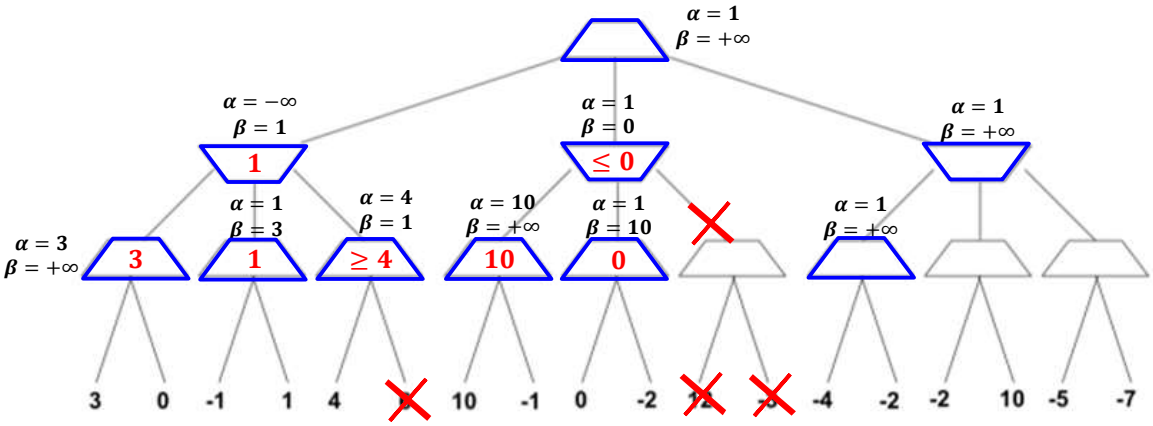
Solution



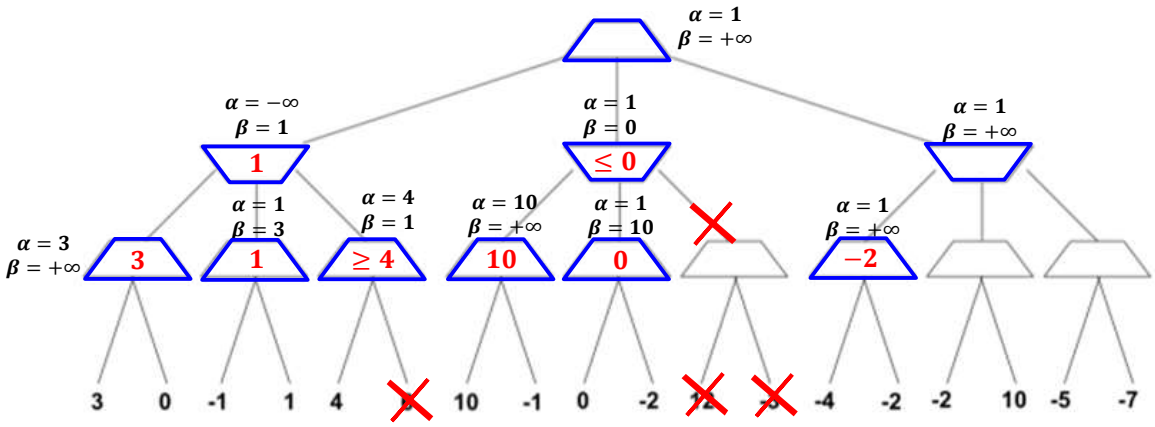
Solution



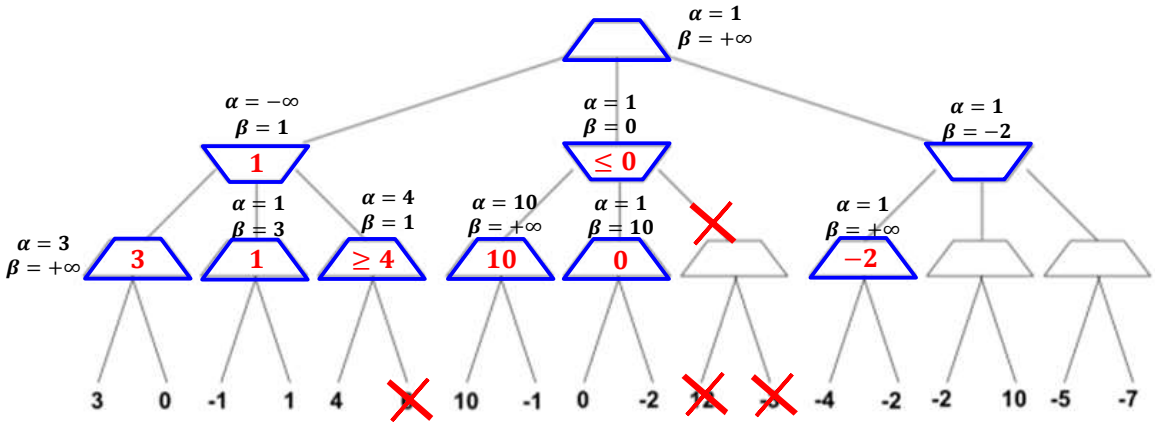
Solution



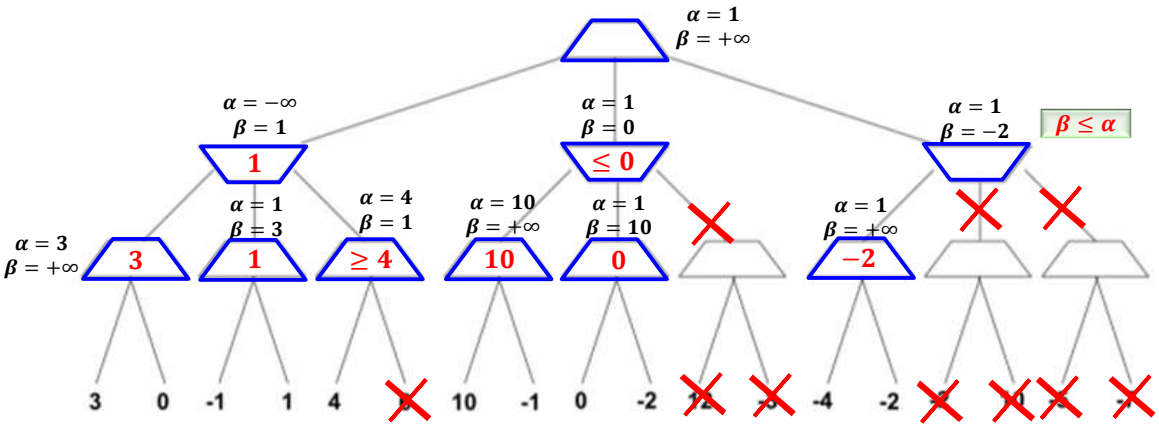
Solution



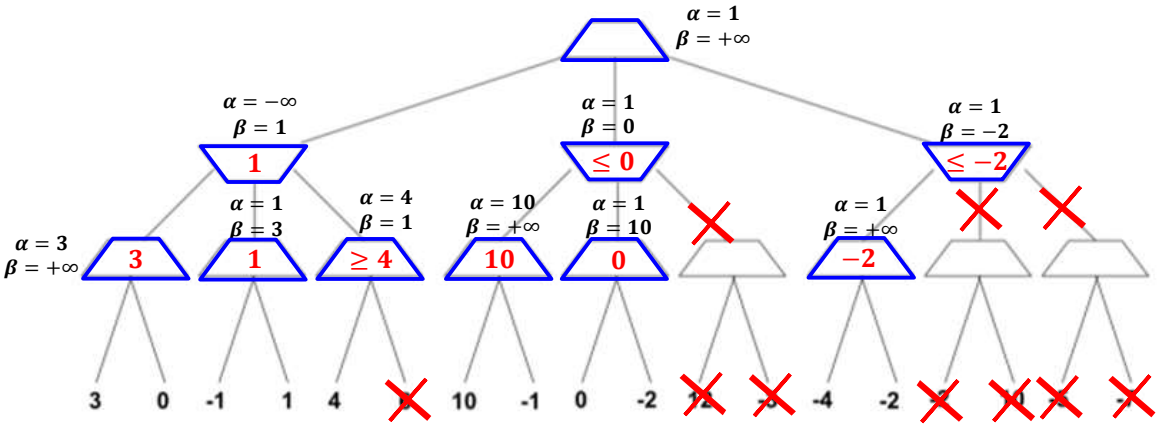
Solution



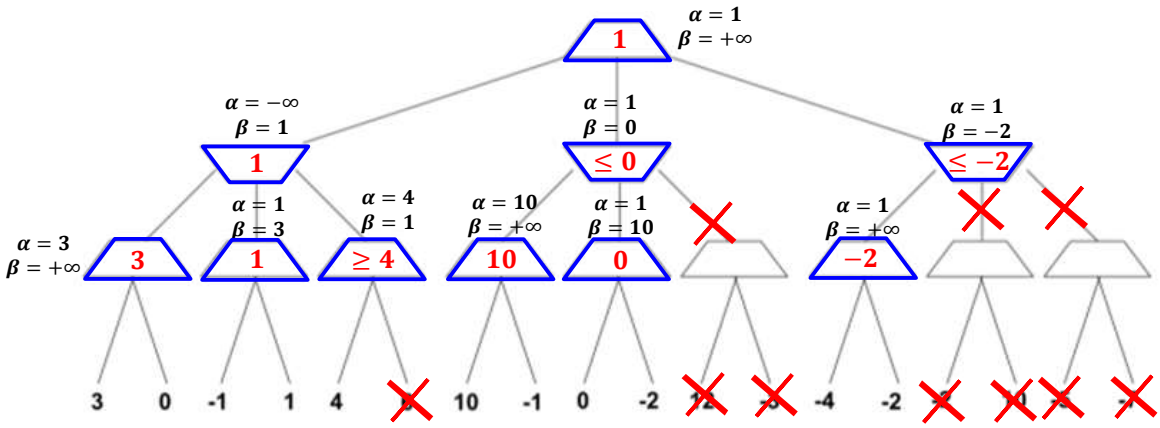
Solution



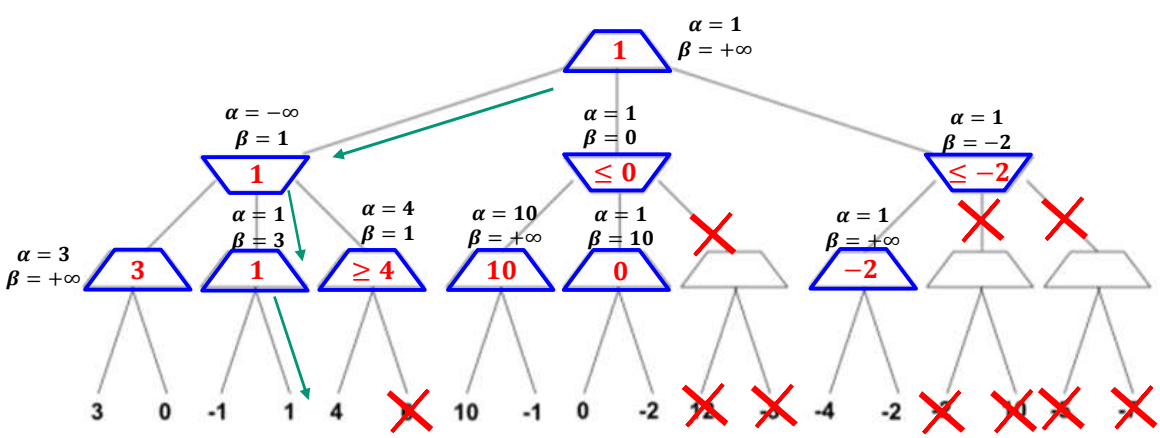
Solution



Solution



Solution



27

Constraint Satisfaction Problems (CSPs)

Slides adapted from Prof. Tuomas Sandholm
Carnegie Mellon University, Computer Science Department

Course Structure/Syllabus

28

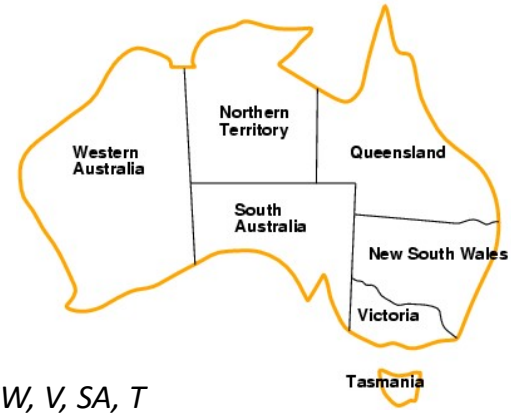
Syllabus	
Timeline	**Topics to be Covered**
Week 1	**INTRODUCTION TO AI:** Artificial Intelligence Introduction, Brief History, Intelligent Agents, Types of agents ***Python Primer for AI***
Week 2-4	**PROBLEM SOLVING BY SEARCH:** Problem formulation, Concept of state space search Introduction to Uninformed Search Techniques: *Breadth First Search, Depth First Search, Depth First Search with Iterative Deepening, Uniform Cost Search* Introduction to Informed/Heuristic search techniques: *Greedy Best First Search, A* search, Hill Climbing search, Simulated Annealing search* Introduction to GA, GA Operations: *Selection, Crossover, Mutation* ***Solving N-Queen Problem using GA*** Adversarial Search: *Minimax algorithm, Alpha-Beta Pruning* ***Building a bot to play tic-tac-toe/Building 8-puzzle solver***
Week 5	**CONSTRAINT SATISFACTION PROBLEM (CSP):** Introduction to CSP, Constraint Graph, Binary and Higher order CSP, Backtracking Search, MRV heuristic, Degree heuristic, Least Constraining-value heuristic, Forward Checking, Arc Consistency, Min-Conflicts Algorithm ***Solving map coloring problem, Solving puzzle***

To be discussed in the first two lectures after the mid-semester exam

Constraint satisfaction problems (CSPs)

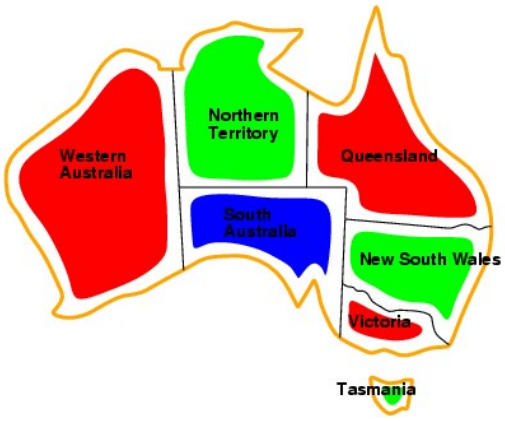
- Standard search problem:
  - **state** is a "black box“ – any data structure that supports successor function and goal test
- CSP:
  - **state** is defined by variables X_i with values from domain D_i
  - A set of constraints specifying allowable combinations of values for subsets of variables
- Allows useful general-purpose algorithms with **more power** than standard search algorithms

Example: Map-Coloring



- Variables: *WA, NT, Q, NSW, V, SA, T*
- Domains: D_i = {red, green, blue}
- Constraints: adjacent regions must have different colors
- e.g., WA ≠ NT, or (WA,NT) in {(red, green),(red, blue),(green, red),(green, blue), (blue, red),(blue, green)}

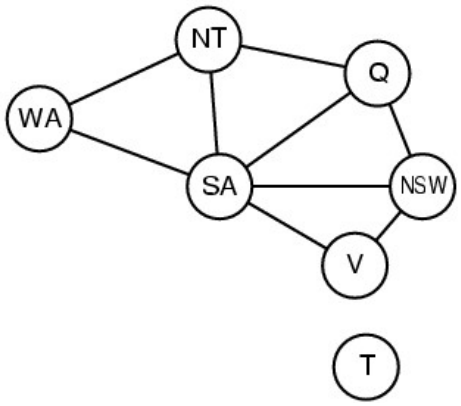
Example: Map-Coloring



- Solutions are complete and consistent assignments
- e.g., WA = red, NT = green, Q = red, NSW = green, V = red, SA = blue, T = green

Constraint graph

- Binary CSP: each constraint relates two variables
- Constraint graph: nodes are variables, arcs are constraints



Varieties of CSPs

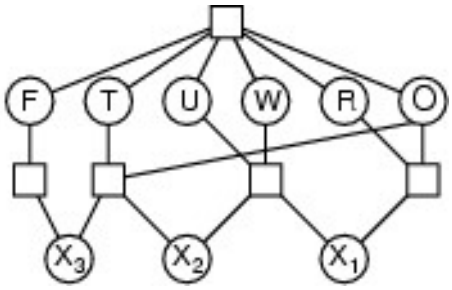
- **Discrete variables**
  - finite domains:
    - n variables, domain size $d \rightarrow O(d^n)$ complete assignments
    - e.g., Boolean CSPs, incl. Boolean satisfiability
  - infinite domains:
    - integers, strings, etc.
    - e.g., job scheduling, variables are start/end days for each job
    - need a constraint language, e.g., $StartJob_1 + 5 \leq StartJob_3$
- **Continuous variables**
  - e.g., start/end times for Hubble Space Telescope observations

Varieties of constraints

- **Unary** constraints involve a single variable,
  - e.g., SA ≠ green
- **Binary** constraints involve pairs of variables,
  - e.g., SA ≠ WA
- **Higher-order** constraints involve 3 or more variables,
  - e.g., cryptarithmetic column constraints

Example: Cryptarithmetic

$$\begin{array}{r} \text{T W O} \\ +\ \text{T W O} \\ \hline \text{F O U R} \end{array}$$



- **Variables:** *F T U W R O* X_1 X_2 X_3
- **Domains:** {*0,1,2,3,4,5,6,7,8,9*}
- **Constraints:** *Alldiff (F,T,U,W,R,O)*

$O + O = R + 10 \cdot X_1$
$X_1 + W + W = U + 10 \cdot X_2$
$X_2 + T + T = O + 10 \cdot X_3$
$X_3 = F,\ T \neq 0,\ F \neq 0$

Questions?