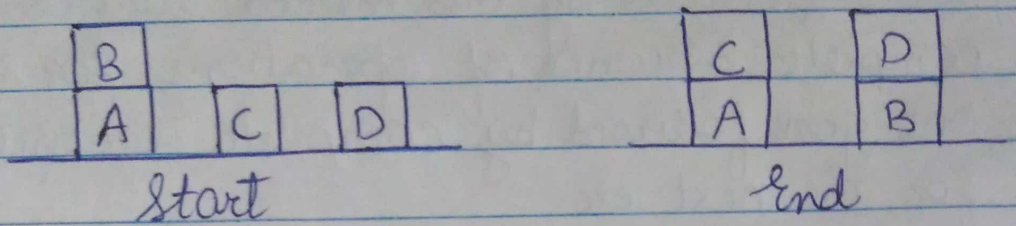
Title : Goal stack planning

Problem statement :

Implement goal stack planning for the following configuration from the block's world.



Objectives :

- To learn and understand concept of goal stack planning
- To study need & use of goal stack planning.
- To implement goal stack planning algorithm using suitable programming language.

Outcomes : We will be able to

- learn the concept of goal stack planning
- study need and use of goal stack planning
- implement goal stack planning.

S/W & H/W : - OS : Ubuntu/Fedora 20 with Python libraries.
requirements

Theory :

- Goal stack planning :
  - One of the earliest techniques in planning uses goal stack.

- Problem solver uses single stack that contains
  - sub goals and operators both.
  - sub goals are solved linearily and then finally the conjoined goal is solved.

- Plans generated by this method will contain complete sequance of operations for solving one goal followed by complete sequance of operations for the next, etc.

- Problem solver relies on:
  - A database that describes the current situation
  - Set of operators with pre-conditions, add & delete lists.

Let us assume that goal to be satisfied is

$GOAL = G1 \mid G2 \mid G3 \ldots G_N$

- Subgoals $G_1, G_2, G_3 \ldots G_N$ are stacked with compound goal.

Top

G_1
G_2
G_3
:
G_N

Bottom

$G_1 \mid G_2 \mid G_3 \ldots \mid G_N$

Algorithm:

1. Find an operator that satisfies sub goal G_1 (makes it true) and replace G_1 by the operator.
   - If more than one operator satisfies subgoals then apply some heuristic to choose one.

2. In order to execute the top most operation, its pre-conditio are added onto the stack.
   - Once the preconditions of an operator can be applied to produce a new state.
   - New state is obtained by using ADD and DELETE lists of an operator to the existing database.

3. Problem solver keep tracks of operations applied.
   - This process is continued till the goal stack is empty and problem solver returns plan of the problem.

Consider given example:

Intial state:

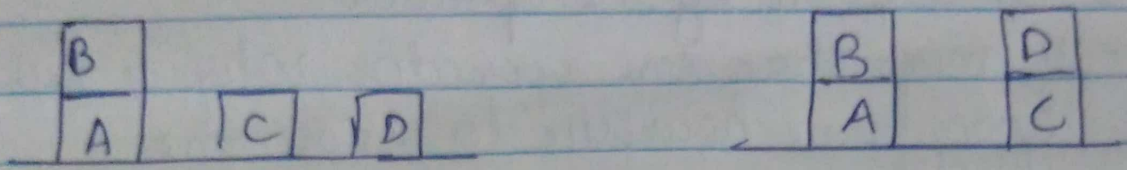
ON(B, A) ^ ONTABLE(C) ⊥ ONTABLE(A) ⊥ ONTABLE(D) ⊥ CLEAR(B) | CLEAR(C) | CLEAR D | ARMEMPTY.

Goal state:

ON(C, A) ⊥ ON(B, D) ⊥ CLEAR(C) ⊥ CLEAR (B) ⊥ ONTABLE(A) ⊥ ONTABLE(D) ⊥ ARMEMPTY.

Test case:

Input



Conclusion:

We have successfully implemented goal stack planning in Python to implement the above case.