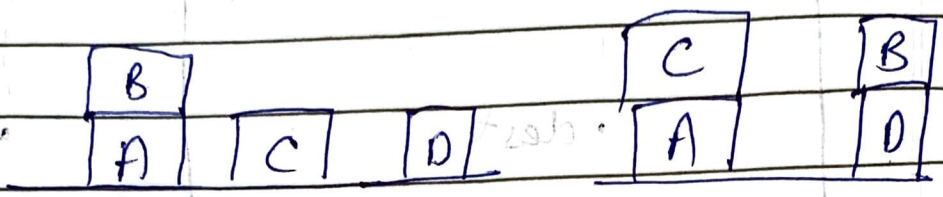
Page No.
Date

Assignment - B3

Title:- Goal Stack planning

Problem Statement:- Implement goal stack planning for the following config. from the blocks.



Start Good

Objective:-

1) To study goal stack planning problem
2) Implement algorithm to solve goal stack planning.

Outcome:-

Understand the concept of goal stack planning

S/w req. & H/w req.:-

1) Python 3
2) 64-bit OS
3) 4GB RAM, 500 GB HDD

Theory:-

Goal Stack planning :-

1) GSP breaks up a set of goal predicates into individual sub goals & attempts to solve them individually one after another.
2) This is also called linear planning.
3) It works by pushing the goal description onto a stack.
4) It pushes both the conjunct as well as each of the individual goal predicates seperately.
5) The algorithms pop the element on top of the stack.
6) If it is a (goal) predicate that is true in the current state and then nothing is done & the next element is popped from the stack.
7) If it is a goal predicate that is not in current state, a relevant action is pushed into the stack, followed by the preconditions, first the conjunctions & then the individual preconditions.
8) The precondition on top of the stack becomes the next subgoal to be addressed recursively

Page No.
Date

Algorithm:- GSP (given state, goal state, actions)

1) State ← given state
2) Plan ← ()
3) Stack ← empty stack
4) Push set (given goal, Stack)
5) While not empty (stack
6) do x ← pop (stack)
7) if x ∈ ← actions
8) then plan ← (plan · x)
9) stack ← progress (x, State)
10) else if x is a conjunct of goal predicate
11) then solvedFlag ← true
12) for each g ∈ c
13) do if g ∈ state
14) then solvedFlag ← False
15) if solvedFlag = FALSE
16) then PushSet (C, Stack)
17) else if x ∈ givenstate
18) then choose action that achieve x
19) if no such action exist
20) then return FALSE
21) PUSH (a, stack)
22) Push set (Precondition (a). stack)
23) return Plan

Conclusion:- We have successfully implemented Goal stack planning.