

## Problem 8: Progression Planning for the Blocks' World Domain

The files `blocksworld.pddl` and `pb3.pddl` define the following PDDL description of the Blocks' World domain.

```
blocksworld.pddl:
(define (domain blocksworld)
  (:requirements :strips :equality)
  (:predicates (clear ?x)
                (on-table ?x)
                (arm-empty)
                (holding ?x)
                (on ?x ?y))

  (:action pickup
    :parameters (?ob)
    :precondition (and (clear ?ob)
                       (on-table ?ob) (arm-empty))
    :effect (and (holding ?ob) (not (clear ?ob))
                 (not (on-table ?ob))
                 (not (arm-empty))))

  (:action putdown
    :parameters (?ob)
    :precondition (and (holding ?ob))
    :effect (and (clear ?ob) (arm-empty)
                 (on-table ?ob)
                 (not (holding ?ob))))

  (:action stack
    :parameters (?ob ?underob)
    :precondition (and (clear ?underob)
                       (holding ?ob))
    :effect (and (arm-empty) (clear ?ob)
                 (on ?ob ?underob)
                 (not (clear ?underob))
                 (not (holding ?ob))))

  (:action unstack
    :parameters (?ob ?underob)
    :precondition (and (on ?ob ?underob)
                       (clear ?ob) (arm-empty))
    :effect (and (holding ?ob) (clear ?underob)
                 (not (on ?ob ?underob))
                 (not (clear ?ob)) (not
                 (arm-empty))))

pb3.pddl
(define (problem pb3)
  (:domain blocksworld)
  (:objects a b c)
  (:init (on-table a) (on-table b)
         (on-table c) (clear a)
         (clear b) (clear c)
         (arm-empty))
  (:goal (and (on a b) (on b c))))
```

Implement progression planning and validate your implementation in the Blocks' world domain. For reading the domain and problem description, use the uploaded PDDL scanner (`pddlscanner.lua`). The function `run()` reads and scans respective PDDL files.

The functions `PDDLScanner:get_domain(ddecl)` and `PDDLScanner:get_problem(pdecl)` yield tables with scanned descriptions. It has the following fields: `name`, `reqs`, `const`, `preds`, `acts` and `pname`, `dname`, `objs`, `init`, `goal`, respectively.

The functions `pretty_print_domain` and `pretty_print_problem` print the read PDDL to the screen and show and show how the data structures are used.