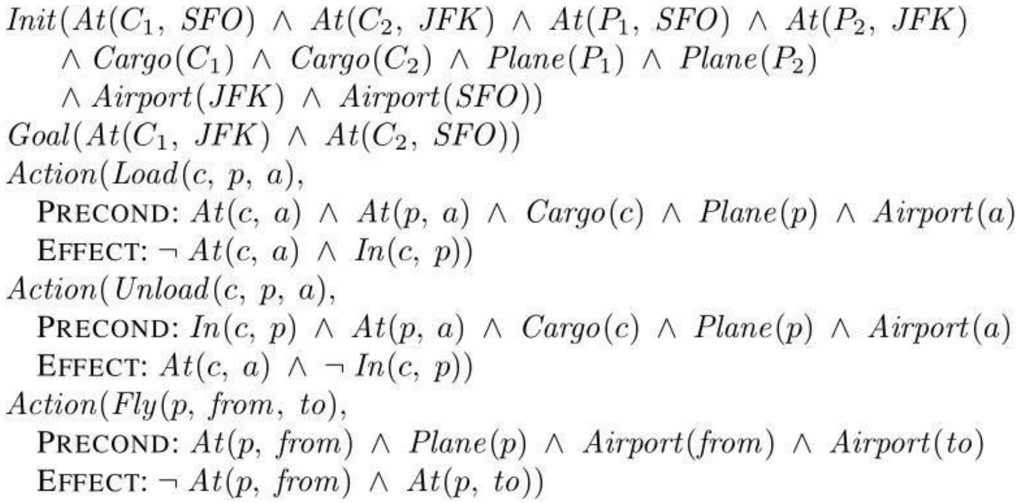
1. Consider the action schemas and initial state below,



$Init(At(C_1,\ SFO) \land At(C_2,\ JFK) \land At(P_1,\ SFO) \land At(P_2,\ JFK)$
$\quad \land Cargo(C_1) \land Cargo(C_2) \land Plane(P_1) \land Plane(P_2)$
$\quad \land Airport(JFK) \land Airport(SFO))$
$Goal(At(C_1,\ JFK) \land At(C_2,\ SFO))$
$Action(Load(c,\ p,\ a),$
$\quad$ PRECOND: $At(c,\ a) \land At(p,\ a) \land Cargo(c) \land Plane(p) \land Airport(a)$
$\quad$ EFFECT: $\neg At(c,\ a) \land In(c,\ p))$
$Action(Unload(c,\ p,\ a),$
$\quad$ PRECOND: $In(c,\ p) \land At(p,\ a) \land Cargo(c) \land Plane(p) \land Airport(a)$
$\quad$ EFFECT: $At(c,\ a) \land \neg In(c,\ p))$
$Action(Fly(p,\ from,\ to),$
$\quad$ PRECOND: $At(p,\ from) \land Plane(p) \land Airport(from) \land Airport(to)$
$\quad$ EFFECT: $\neg At(p,\ from) \land At(p,\ to))$

(a) What are the applicable actions that are instances of $Fly(p, from, to)$ in the state described by

$At(P_1, JFK) \land At(P_2, SFO) \land Plane(P\,1) \land Plane(P_2) \land Airport(JFK) \land Airport(SFO)$?

and what are the corresponding progressed states?

(b) What are the relevant actions that are instances of $Fly(p, from, to)$ in the goal

$$At(C_1, JFK) \land At(C_2, SFO)$$

and what are the corresponding regressed subgoals?

Answer.

(a) Applicable actions:

- $Fly(P_1, JFK, SFO)$: $At(P_2, SFO) \land Plane(P\,1) \land Plane(P_2) \land Airport(JFK) \land Airport(SFO) \land At(P_1, SFO)$
- $Fly(P_2, SFO, JFK)$ $At(P_1, JFK) \land Plane(P\,1) \land Plane(P_2) \land Airport(JFK) \land Airport(SFO) \land At(P_2, JFK)$

Further applicable (but inefficient) actions given that there is no requirement that the airports be different:

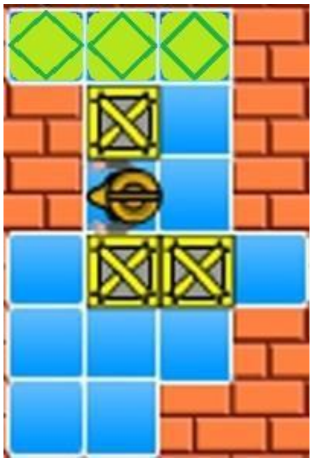
- $Fly(P_1, JFK, JFK)$: $At(P_1, JFK) \land At(P_2, SFO) \land Plane(P\,1) \land Plane(P_2) \land Airport(JFK) \land Airport(SFO)$
- $Fly(P_2, SFO, SFO)$ $At(P_1, JFK) \land At(P_2, SFO) \land Plane(P\,1) \land Plane(P_2) \land Airport(JFK) \land Airport(SFO)$

(b) Relevant actions:

- $Fly(C_2, JFK, SFO)$: $At(C_2, JFK) \wedge Plane(C_2) \wedge Airport(JFK) \wedge Airport(SFO) \wedge At(C_1, JFK)$
- $Fly(C_1, SFO, JFK)$: $At(C_1, SFO) \wedge Plane(C_1) \wedge Airport(JFK) \wedge Airport(SFO) \wedge At(C_2, SFO)$
- $Fly(C_2, y, SFO)$: $At(C_2, y) \wedge Plane(C_2) \wedge Airport(y) \wedge Airport(SFO) \wedge At(C_1, JFK)$
- $Fly(C_1, y, JFK)$: $At(C_1, y) \wedge Plane(C_1) \wedge Airport(JFK) \wedge Airport(y) \wedge At(C_2, SFO)$

2. Sokoban is a puzzle game in which the player has to move around boxes so that they reach a certain place. The rules are that the player can move horizontally or vertically as long as there are no boxes or bricks, and can push a box as long as the player is in front of it and there is space for the box to move. However, the player cannot push two boxes at once. All the boxes have to reach the target (represented as diamonds in the drawing). Solve this problem as a planning problem by writing a PDDL description of the domain. Also write a PDDL description of the problem instance shown in the drawing, and solve it on http://editor.planning.domains/.

Hint: Represent each location as a pair of coordinates (x, y) where x, y are integers. Then use two relations *inc* and *dec* to encode successor relations on integers, e.g., $inc(1, 2), inc(2, 3), inc(3, 4), \ldots, dec(2, 1), dec(3, 2), dec(4, 3), \ldots$



Answer: Below is a part of the domain description. Full solution to be downloaded from Canvas.

```
(:predicates (wall ?x ?y) (box ?x ?y) (at ?x ?y) (inc ?p ?pp) (dec ?pp ?p))


(:action  move-up
     :parameters (?x ?y ?xn)
     :precondition (and (at ?x ?y) (not (wall ?xn ?y)) (not (box ?xn ?y)) (dec ?x ?xn))
     :effect (and (not (at ?x ?y)) (at ?xn ?y))
)

 (:action  push-down
     :parameters (?x ?y ?xn ?xnn)
     :precondition (and (at ?x ?y) (not (wall ?xn ?y)) (box ?xn ?y) (inc ?x ?xn)
       (not (wall ?xnn ?y)) (not (box ?xnn ?y)) (inc ?xn ?xnn))
     :effect (and (not (at ?x ?y)) (at ?xn ?y) (not (box ?xn ?y)) (box ?xnn ?y))
)
```