Solutions to
Exercises Lecture 12
Efficient AI Programming (IAIP)

Exercise 1 (adapted from RN03 11.2)

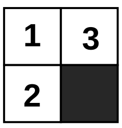
Given the STRIPS problem defined in Figure 11.2 on page 380 in RN03,

a) What are all the applicable concrete instances of *Fly*(*p*,*from*,*to*) in the state *At*(P1,JFK) ∧ *At*(P2,SFO) ∧ *Plane*(P1) ∧ *Plane*(P2) ∧ *Airport*(JFK) ∧ *Airport*(SFO)?
b) What is the next state achieved by applying each of these actions?

Applicable Action	**Next State**
Fly(P1,JFK,SFO)	*At*(P1,SFO) ∧ *At*(P2,SFO) ∧ *Plane*(P1) ∧ *Plane*(P2) ∧ *Airport*(JFK) ∧ *Airport*(SFO)
Fly(P1,JFK,JFK)	*At*(P1,JFK) ∧ *At*(P2,SFO) ∧ *Plane*(P1) ∧ *Plane*(P2) ∧ *Airport*(JFK) ∧ *Airport*(SFO)
Fly(P2,SFO,JFK)	*At*(P1,JFK) ∧ *At*(P2,JFK) ∧ *Plane*(P1) ∧ *Plane*(P2) ∧ *Airport*(JFK) ∧ *Airport*(SFO)
Fly(P2,SFO,SFO)	*At*(P1,JFK) ∧ *At*(P2,SFO) ∧ *Plane*(P1) ∧ *Plane*(P2) ∧ *Airport*(JFK) ∧ *Airport*(SFO)

Exercise 2

Consider the 4-puzzle domain shown below



As for the 8-puzzle, tiles can be moved up, down, left, and right, but it is only possible to move a tile into an empty space. Thus, for the state shown above, tile 3 can move down and tile 2 can move right.

a) Define the 4-puzzle as a STRIPS planning domain.

This can be done in many ways. Below is one suggestion:

Objects:
1,2,3,Empty

Predicates:
At(*t*,*r*,*c*) : Tile *t* is a row *r* and column *c*. Top-most row is 1. Left-most column is 1
Pos(*n*) : *n* is a position index (1 or 2)
Tile(*t*) : *t* is a tile

Actions schemas:

Up(*t*,*c*)
PRE: *At*(*t*,2,*c*) ∧ *At*(Empty,1,*c*) ∧ *Pos*(*c*) ∧ *Tile*(*t*)
EFF: ¬*At*(*t*,2,*c*) ∧ *At*(*t*,1,*c*) ∧ ¬*At*(Empty,1,*c*) ∧ *At*(Empty,2,*c*)

Down(*t*,*c*)
PRE: *At*(*t*,1,*c*) ∧ *At*(Empty,2,*c*) ∧ *Pos*(*c*) ∧ *Tile*(*t*)
EFF: ¬*At*(*t*,1,*c*) ∧ *At*(*t*,2,*c*) ∧ ¬*At*(Empty,2,*c*) ∧ *At*(Empty,1,*c*)

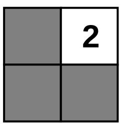
Right(*t*,*r*)
PRE: *At*(*t*,*r*,1) ∧ *At*(Empty,r,2) ∧ *Pos*(*r*) ∧ *Tile*(*t*)
EFF: ¬*At*(*t*,*r*,1) ∧ *At*(*t*,*r*,2) ∧ ¬*At*(Empty,*r*,2) ∧ *At*(Empty,*r*,1)

Left(*t*,*r*)
PRE: *At*(*t*,*r*,2) ∧ *At*(Empty,r,1) ∧ *Pos*(*r*) ∧ *Tile*(*t*)
EFF: ¬*At*(*t*,*r*,2) ∧ *At*(*t*,*r*,1) ∧ ¬*At*(Empty,*r*,1) ∧ *At*(Empty,*r*,2)

b) Assume that the above state is the initial state. How would it be represented for your domain description?

At(1,1,1) ∧ *At*(2,2,1) ∧ *At*(3,1,2) ∧ *At*(Empty,2,2) ∧ *Pos*(1) ∧ *Pos*(2) ∧ *Tile*(1) ∧ *Tile*(2) ∧ *Tile*(3)

Assume the goal is to place tile 2 in the top right corner



c) How would this goal be represented for your domain description?

At(2,1,2)

d) What are the applicable actions of the goal for a regression search?

e) What is the new goal for each action?

Action	New Goal
Up(2,2)	$At(2,2,2) \wedge At(\text{Empty},1,2) \wedge Pos(2) \wedge Tile(2)$
Right(2,1)	$At(2,1,1) \wedge At(\text{Empty},1,2) \wedge Pos(1) \wedge Tile(2)$

Exercise 3

Consider the STRIPS planning problem

Init = *v*

Goal = v $\wedge$ w

Action *A*
PRE: *v*
EFF: *w*

Action *B*
PRE:
EFF: $\neg v$

a) Write a propositional formula that is satisfiable if and only if a plan exists in one time step.

Expr $\equiv$ Init $\wedge$ Goal $\wedge$ Vars $\wedge$ Pre $\wedge$ Cons

Init $\equiv v^0 \wedge \neg w^0$

Goal $\equiv v^1 \wedge w^1$

Vars $\equiv [v^1 \Leftrightarrow v^0 \wedge \neg(v^0 \wedge B^0)] \quad \wedge$
$[w^1 \Leftrightarrow w^0 \vee (\neg w^0 \wedge A^0)]$

Pre $\equiv A^0 \Rightarrow v^0$

Cons $\equiv \neg (A^0 \wedge B^0)$

b) Show a variable assignment that makes the formula true.

v^0 = True, w^0 = False, A^0 = True, B^0 = False, v^1 = True, w^1 = True