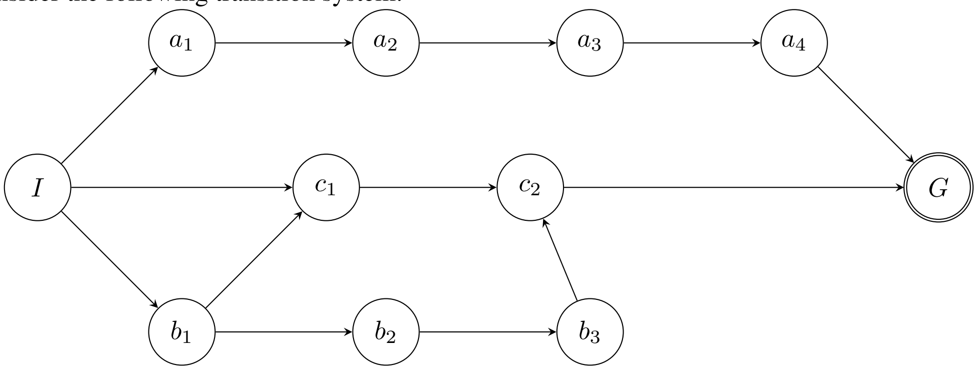
Exercise 1

Consider the following transition system:



(i) For the following abstraction functions, draw the abstract state space, and compute the heuristic value h^α for the initial state I.

- α_1

I	a_1	a_2	a_3	a_4	b_1	b_2	b_3	c_1	c_2	G
s_1	s_2	s_3	s_4	s_5	s_1	s_2	s_3	s_5	s_1	s_5

- α_2

I	a_1	a_2	a_3	a_4	b_1	b_2	b_3	c_1	c_2	G
s_1	s_2	s_3	s_1	s_2	s_3	s_1	s_2	s_1	s_2	s_3

- α_3

I	a_1	a_2	a_3	a_4	b_1	b_2	b_3	c_1	c_2	G
s_1	s_1	s_2	s_3	s_4	s_1	s_2	s_3	s_2	s_3	s_4

(ii) Which of the abstractions is preferable? Discuss the advantages and disadvantages of each of them.

(iii) Run A* search using the heuristic provided by α_3. Draw the corresponding search tree, annotating each node with its g and h values. How many nodes are expanded until finding the optimal plan?

Exercise 2

Consider the following transportation problem where one DHL Xmas truck has to transport two gifts to their goal location. At any point in time only one package can be loaded into the truck (There are 2 big gifts and a small Express DHL Truck). Formally, this task is encoded as the following FDR planning task $\Pi = (V, A, I, G)$ where

- $V = \{t, g_1, g_2, c\}$ with domains
  - $D(t) = \{1, 2, 3\}$,
  - $D(g_1) = D(g_2) = \{1, 2, 3, T\}$,
  - $D(c) = \{0, 1\}$ (c is the capacity of the DHL Truck)
- $A = \{drive(x, y), load(i, z), unload(i, z)\}$ for $\{x, y\} \in \{\{1, 2\}, \{2, 3\}\}$, $i \in \{1, 2\}$, and $z \in \{1, 2, 3\}$
  - $pre(drive(x, y)) = \{t = x\}$,
    $eff(drive(x, y)) = \{t = y\}$
  - $pre(load(i, z)) = \{t = z, g_i = z, c = 1\}$,
    $eff(load(i, z)) = \{g_i = T, c = 0\}$
  - $pre(unload(i, z)) = \{t = z, g_i = T, c = 0\}$,
    $eff(unload(i, z)) = \{g_i = z, c = 1\}$
- $I = \{t = 1, g_1 = 2, g_2 = 3, c = 1\}$,
- $G = \{t = 1, g_1 = 3, g_2 = 2\}$.

All actions have unit-cost.
Consider the patterns $P_1 = \{t\}$, $P_2 = \{g_1, c\}$, and $P_3 = \{t, g_2\}$.

(i) Compute a pattern database for each of the three patterns. To do so, execute the algorithm given in the lecture: Construct the reachable state space $\Theta_\Pi^{\pi_{P_i}}$ of the syntactic projection onto P_i by a breadth-first forward search. Write up the states, annotating them by their variable values (e.g., the initial state of $\Theta_\Pi^{\pi_{P_3}}$ can be notated $t2g_23$), and draw an edge from each state to each of its successor states. Annotate each of the states in $\Theta_\Pi^{\pi_{P_i}}$ with its goal distance.

(ii) Create the perfect hash function and corresponding look-up table for the pattern database for P_3. Do so using the method given in the lecture, setting $v_1 = t$ and $v_2 = g_2$ with the correspondences $D_t = \{1, 2, 3\} \approx \{0, 1, 2\}$ and $D_{g_2} = \{1, 2, 3, T\} \approx \{0, 1, 2, 3\}$. Give the result in terms of the final table containing, as in the lecture slides, one row with the states, one row with their hash values, and one row with the corresponding heuristic values.

Exercise 3

Consider a Blocksworld task with two blocks A, B with initial state and goal as shown in Figure 1.

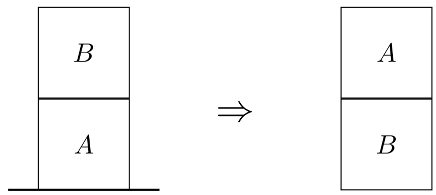


Figure 1: Initial (left) and goal state (right) of the example for Exercise 5.

The encoding has three types of variables, $on(x)$ to describe which block is on top of block x (so on "points upwards"); $ontable(x)$, to say if a block is currently on the table, and $holding$, describing which block is currently held in the hand. The $\perp$ symbol represents the fact that *nothing* is on top of a block, respectively *nothing* is held in the hand.

There are four types of actions, $(un)stack(x, y)$, to $(un)stack$ block x onto/away from block y; $pickup(x)$ and $drop(x)$ to $pickup$ block x from the table into the hand, or $drop$ it to the table if it is held in the hand. In the initial state depicted in Figure 1, the only applicable action is $unstack(A, B)$.

Formally, the FDR encoding of the task, $\Pi = (V, A, I, G)$, is defined as follows:

- Variables: $V = \{on(x), holding, ontable(x)\}$ for $x \in \{A, B\}$ with domains
  - $D(ontable(x)) = \{\top, \perp\}$,
  - $D(on(x)) = \{A, B, \perp\} \setminus \{x\}$, and
  - $D(holding) = \{A, B, \perp\}$.
- Actions: $A = \{stack(x, y), unstack(x, y), pickup(x), drop(x)\}$ (uniform action costs)
  - $stack(x, y)$, with $x \neq y \in \{A, B\}$
    pre: $\{holding = x, on(y) = \perp\}$,
    eff: $\{holding = \perp, on(y) = x\}$
  - $unstack(x, y)$, with $x \neq y \in \{A, B\}$
    pre: $\{holding = \perp, on(y) = x, on(x) = \perp\}$,
    eff: $\{holding = x, on(y) = \perp\}$
  - $pickup(x)$, with $x \in \{A, B\}$
    pre: $\{holding = \perp, ontable(x) = \top, on(x) = \perp\}$,
    eff: $\{holding = x, ontable(x) = \perp\}$
  - $drop(x)$, with $x \in \{A, B\}$
    pre: $\{holding = x\}$,
    eff: $\{holding = \perp, ontable(x) = \top\}$
- Initial state:$I = \{on(A) = B, on(B) = \perp, holding = \perp, ontable(A) = \top, ontable(B) = \perp\}$
- Goal: $G = \{on(B) = A\}$

Compute a pattern database for the pattern $P_1 = \{holding, on(B)\}$. To do so, execute the algorithm given in the lecture: Construct the abstract state space (drawing it) Θ' of the syntactic projection onto P_1. Denote the states by their variable values (e.g., the abstract state $s = \{holding = \perp, on(B) = A\}$ can be denoted $\perp A$), and draw an edge from each state to each of its successor states. Annotate all states in Θ' with their goal distance; annotate each transition by the labels that induce the transition. What is the heuristic value for the initial state for P_1?

Exercise 4

Is there an overlap between the actions used by the optimal solution in the abstract state space and the ones in the optimal plan for the original planning task.

Show that this is not the case, by providing a planning task, and a pattern ignoring a single variable, in which the optimal plan is not empty but completely different from the optimal plan in the original state space.

Hint: Use action costs to more easily control what is the optimal plan.