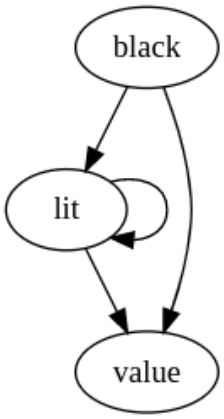
ˆ Explain the rationale behind the PDDL representation you have chosen for the problem. [12.5 marks]
PDDL is a standard for defining problems that existing planners solve. As the problem we are introduced to requires a step by step plan for a solution and each step is discrete, PDDL 1.2 (version used) is a good standard for modelling the problem. I could have used a custom implementation which would have perhaps made the simulation of the world easier, however this would not have been compatible with existing planners.

ˆ Draw the causal graph for this problem. [12.5 marks]



ˆ Show tables with the runtime and number of expanded nodes of each heuristic for solving both problems with A* (w=1). [12.5 marks]
P01:

```
f = 1 [1 evaluated, 0 expanded, t=0s, 6236 KB]
Best heuristic value: 1 [g=0, 1 evaluated, 0 expanded, t=0s, 6236 KB]
f = 2 [20 evaluated, 1 expanded, t=0s, 6236 KB]
f = 3 [142 evaluated, 20 expanded, t=0s, 6236 KB]
f = 4 [470 evaluated, 142 expanded, t=0s, 6236 KB]
f = 5 [851 evaluated, 470 expanded, t=0s, 6236 KB]
f = 6 [1045 evaluated, 851 expanded, t=0s, 6236 KB]
```

P02:

```
f = 1 [1 evaluated, 0 expanded, t=0s, 6236 KB]
Best heuristic value: 1 [g=0, 1 evaluated, 0 expanded, t=0s, 6236 KB]
f = 2 [21 evaluated, 1 expanded, t=0s, 6236 KB]
f = 3 [154 evaluated, 21 expanded, t=0s, 6236 KB]
f = 4 [503 evaluated, 154 expanded, t=0s, 6236 KB]
f = 5 [835 evaluated, 503 expanded, t=0s, 6236 KB]
f = 6 [946 evaluated, 835 expanded, t=0s, 6236 KB]
```

ˆ Using k = 1, 10, 20, 30, 40, 50, 60, 70, 80, 90 and 100 as the weight of the WA* algorithm, show the length and number of expanded nodes for both the blind and goalcount heuristics for solving both problems. Which is your optimal solution length? What is your longest suboptimal solution?. [12.5 marks]
Length &expanded nodes

k	1	10	20	30	40	50	60	70	80	90	100
b01	997,6	997,6	997,6	997,6	997,6	997,6	997,6	997,6	997,6	997,6	997,6
gc01	804,6	804,6	804,6	804,6	804,6	804,6	804,6	804,6	804,6	804,6	804,6
b02	907,6	907,6	907,6	907,6	907,6	907,6	907,6	907,6	907,6	907,6	907,6
gc02	628,6	628,6	628,6	628,6	628,6	628,6	628,6	628,6	628,6	628,6	628,6

ˆ Why does the blind heuristic have the same number of expanded nodes regardless of the chosen weight for the WA* algorithm?. [12.5 marks] For the extension:
Astar is f=g+h
wAstar is f = g + wh

W is a bias applied onto the heuristic, this is done to reduce the search time (with increasing w)
Hence solutions are not optimal.

As df/dw = h
When h is 0 (blind heuristic) there is no change in f with respect to w.
Hence w has no effect on the wastar (f) when using the blind heuristic.

ˆ Explain the rationale behind the construction of the h 1 and h 2 heuristics. [12.5 marks]
It is the way to always find the optimal path to a subgoal.
1 and 2 refer to the number of subgoals to consider <m (total subgoals).
Computation increases exponentially with m, so In practice we use m=1,2,3.

ˆ Show tables with the runtime and number of expanded nodes of h 1 and h 2 heuristics for solving both problems with A* (w=1). [12.5 marks]
I did not manage to finish implementing the heuristic, due to issues I had with my implementation. I would have followed the logic described int he slides and in lab 7. .

ˆ Discuss which is the best heuristic (in terms of number of expanded nodes) among of the heuristics you have implemented when using the A* algorithm (w=1). [12.5 marks]

The best heuristics in terms of fewest expanded nodes should be h2 followed by h1, goal_count and finally blind.