Agent	Pit	Gold	
		Wumpus	
		Gold3	
	Gold2	Pit2	

Wumpus-problem-3.pddl

```
(define (problem wumpus-problem-2)
    (:domain wumpus-domain)
    (:objects sq-1-1 sq-1-2 sq-1-3 sq-1-4 sq-2-1 sq-2-2 sq-2-3 sq-2-4 sq-3-1 sq-3-2 sq-3-3 sq-3-4 sq-4-1 sq-4-2 sq-4-3 sq-4-4 the-gold the-gold2 the-gold3 the-arrow agent wumpus)
    (:init
        (adj sq-1-1 sq-1-2) (adj sq-1-2 sq-1-1)
        (adj sq-1-1 sq-2-1) (adj sq-2-1 sq-1-1)
        (adj sq-1-2 sq-1-3) (adj sq-1-3 sq-1-2)
        (adj sq-1-2 sq-2-2) (adj sq-2-2 sq-1-2)
        (adj sq-1-3 sq-1-4) (adj sq-1-4 sq-1-3)
        (adj sq-1-3 sq-2-3) (adj sq-2-3 sq-1-3)
        (adj sq-1-4 sq-2-4) (adj sq-2-4 sq-1-4)

        (adj sq-2-1 sq-2-2) (adj sq-2-2 sq-2-1)
        (adj sq-2-1 sq-3-1) (adj sq-3-1 sq-2-1)
        (adj sq-2-2 sq-2-3) (adj sq-2-3 sq-2-2)
        (adj sq-2-2 sq-3-2) (adj sq-3-2 sq-2-2)
        (adj sq-2-3 sq-2-4) (adj sq-2-4 sq-2-3)
        (adj sq-2-3 sq-3-3) (adj sq-3-3 sq-2-3)
        (adj sq-2-4 sq-3-4) (adj sq-3-4 sq-2-4)

        (adj sq-3-1 sq-3-2) (adj sq-3-2 sq-3-1)
        (adj sq-3-1 sq-4-1) (adj sq-4-1 sq-3-1)
        (adj sq-3-2 sq-3-3) (adj sq-3-3 sq-3-2)
        (adj sq-3-2 sq-4-2) (adj sq-4-2 sq-3-2)
        (adj sq-3-3 sq-3-4) (adj sq-3-4 sq-3-3)
        (adj sq-3-3 sq-4-3) (adj sq-4-3 sq-3-3)
        (adj sq-3-4 sq-4-4) (adj sq-4-4 sq-3-4)

        (adj sq-4-1 sq-4-2) (adj sq-4-2 sq-4-1)
        (adj sq-4-2 sq-4-3) (adj sq-4-3 sq-4-2)
        (adj sq-4-3 sq-4-4) (adj sq-4-4 sq-4-3)


        (pit sq-1-2)
        (pit sq-4-3)
        (is-gold the-gold)
        (at the-gold sq-1-3)
        (is-gold the-gold2)
        (at the-gold2 sq-3-3)
        (is-gold the-gold3)
        (at the-gold3 sq-4-2)
        (is-agent agent)
        (at agent sq-1-1)
        (is-arrow the-arrow)
        (have agent the-arrow)
        (is-wumpus wumpus)
        (at wumpus sq-2-3)
        (wumpus-in sq-2-3)
    )
    (:goal (and (have agent the-gold) (at agent sq-1-1)))
)
```

1)

(move-wumpus wumpus sq-2-3 sq-2-4)

(move-agent agent sq-1-1 sq-2-1)

(move-agent agent sq-2-1 sq-2-2)

(move-agent agent sq-2-2 sq-2-3)

(move-agent agent sq-2-3 sq-1-3)

(take agent the-gold sq-1-3)

(move-agent agent sq-1-3 sq-2-3)

(move-agent agent sq-2-3 sq-2-2)

(move-agent agent sq-2-2 sq-2-1)

(move-agent agent sq-2-1 sq-1-1)

```
(:action move-wumpus
  :parameters (wumpus sq-2-3 sq-2-4)
  :precondition
    (and
      (is-wumpus wumpus)
      (at wumpus sq-2-3)
      (adj sq-2-3 sq-2-4)
      (not
        (pit sq-2-4)
      )
      (not
        (wumpus-in sq-2-4)
      )
    )
  :effect
    (and
      (not
        (at wumpus sq-2-3)
      )
      (at wumpus sq-2-4)
      (not
        (wumpus-in sq-2-3)
      )
      (wumpus-in sq-2-4)
    )
)
```

2)

(move-wumpus wumpus sq-2-3 sq-2-4)

(move-agent agent sq-1-1 sq-2-1)

(move-agent agent sq-2-1 sq-2-2)

(move-agent agent sq-2-2 sq-2-3)

(move-agent agent sq-2-3 sq-1-3)

(take agent the-gold sq-1-3)

(move-agent agent sq-1-3 sq-2-3)

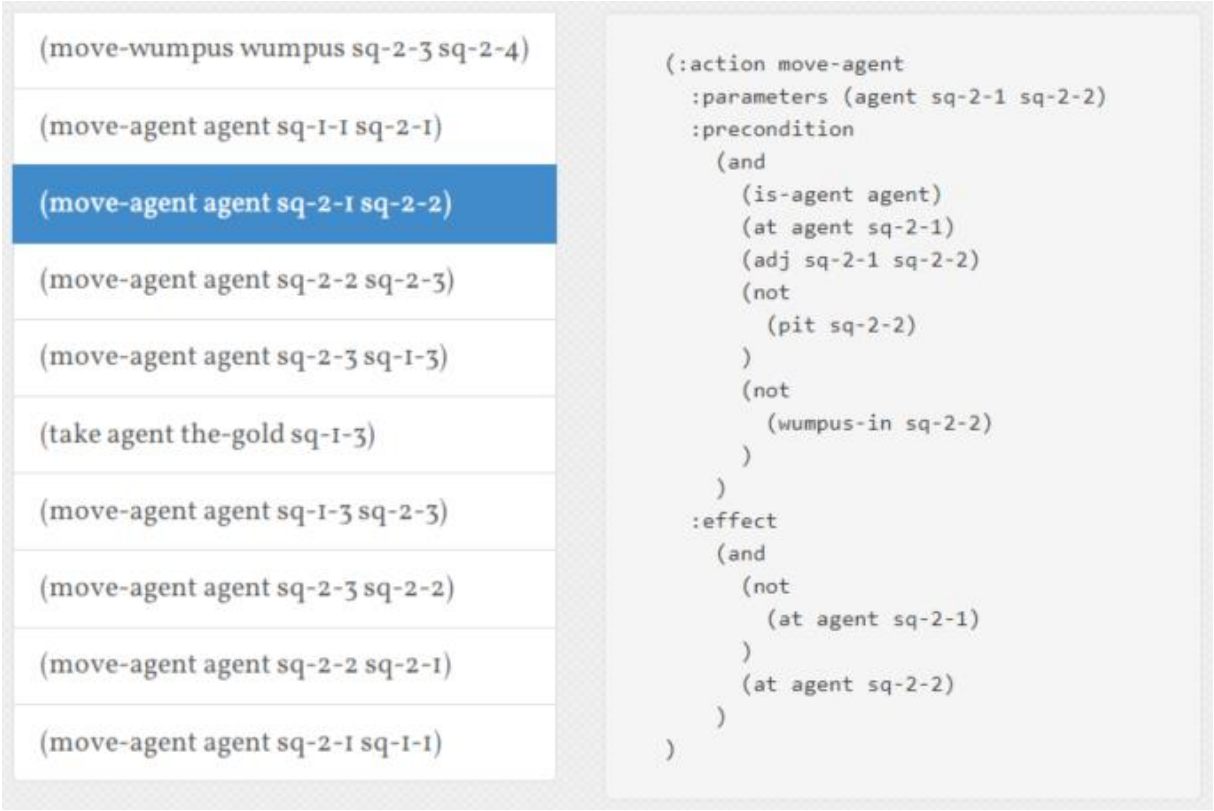
(move-agent agent sq-2-3 sq-2-2)

(move-agent agent sq-2-2 sq-2-1)

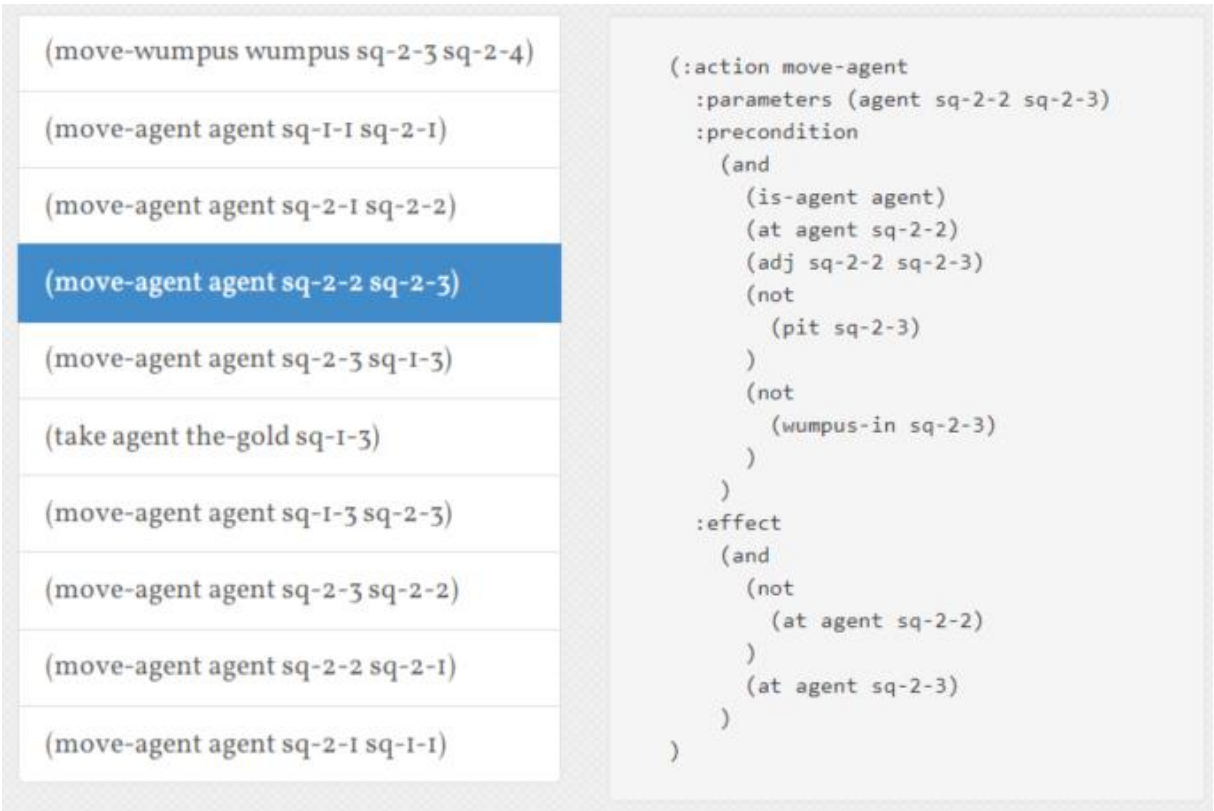
(move-agent agent sq-2-1 sq-1-1)

```
(:action move-agent
  :parameters (agent sq-1-1 sq-2-1)
  :precondition
    (and
      (is-agent agent)
      (at agent sq-1-1)
      (adj sq-1-1 sq-2-1)
      (not
        (pit sq-2-1)
      )
      (not
        (wumpus-in sq-2-1)
      )
    )
  :effect
    (and
      (not
        (at agent sq-1-1)
      )
      (at agent sq-2-1)
    )
)
```

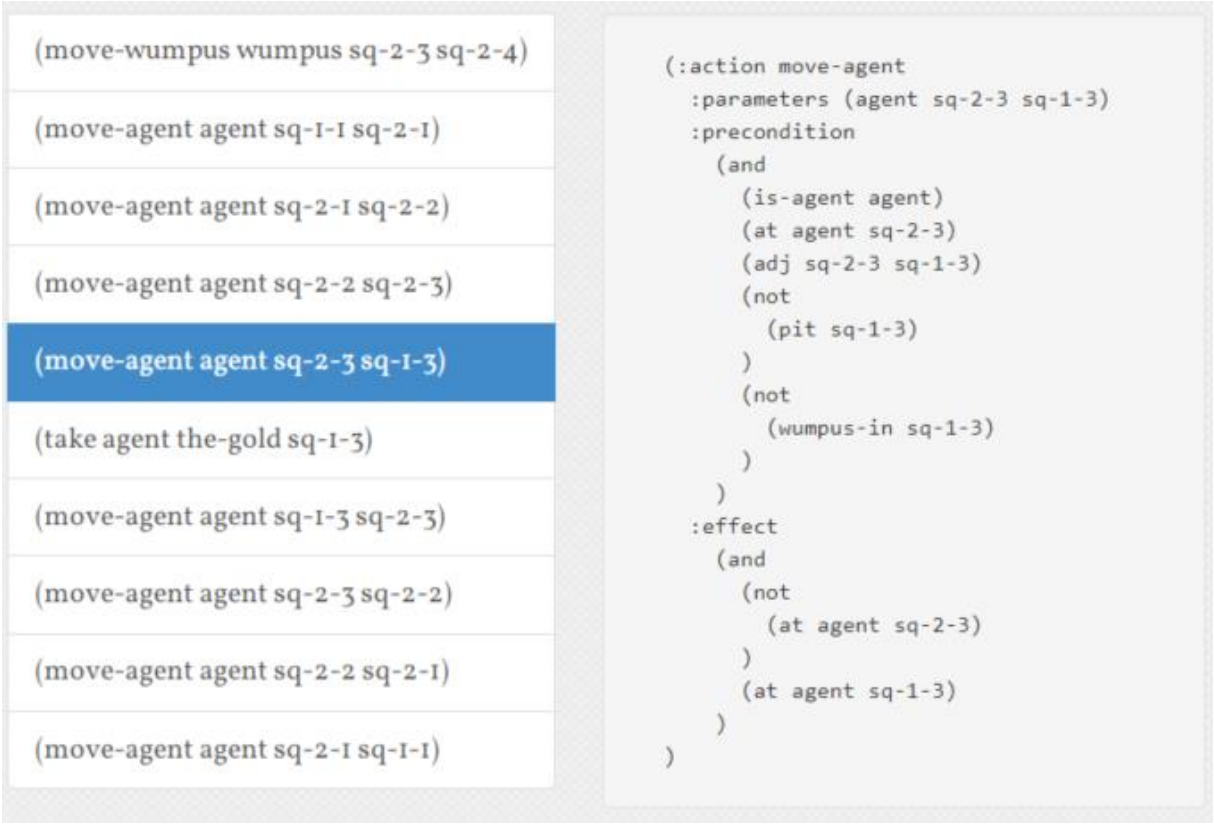
3)


(move-wumpus wumpus sq-2-3 sq-2-4)
(move-agent agent sq-1-1 sq-2-1)
(move-agent agent sq-2-1 sq-2-2)
(move-agent agent sq-2-2 sq-2-3)
(move-agent agent sq-2-3 sq-1-3)
(take agent the-gold sq-1-3)
(move-agent agent sq-1-3 sq-2-3)
(move-agent agent sq-2-3 sq-2-2)
(move-agent agent sq-2-2 sq-2-1)
(move-agent agent sq-2-1 sq-1-1)
(:action move-agent
:parameters (agent sq-2-1 sq-2-2)
:precondition
(and
(is-agent agent)
(at agent sq-2-1)
(adj sq-2-1 sq-2-2)
(not
(pit sq-2-2)
)
(not
(wumpus-in sq-2-2)
)
)
:effect
(and
(not
(at agent sq-2-1)
)
(at agent sq-2-2)
)
)

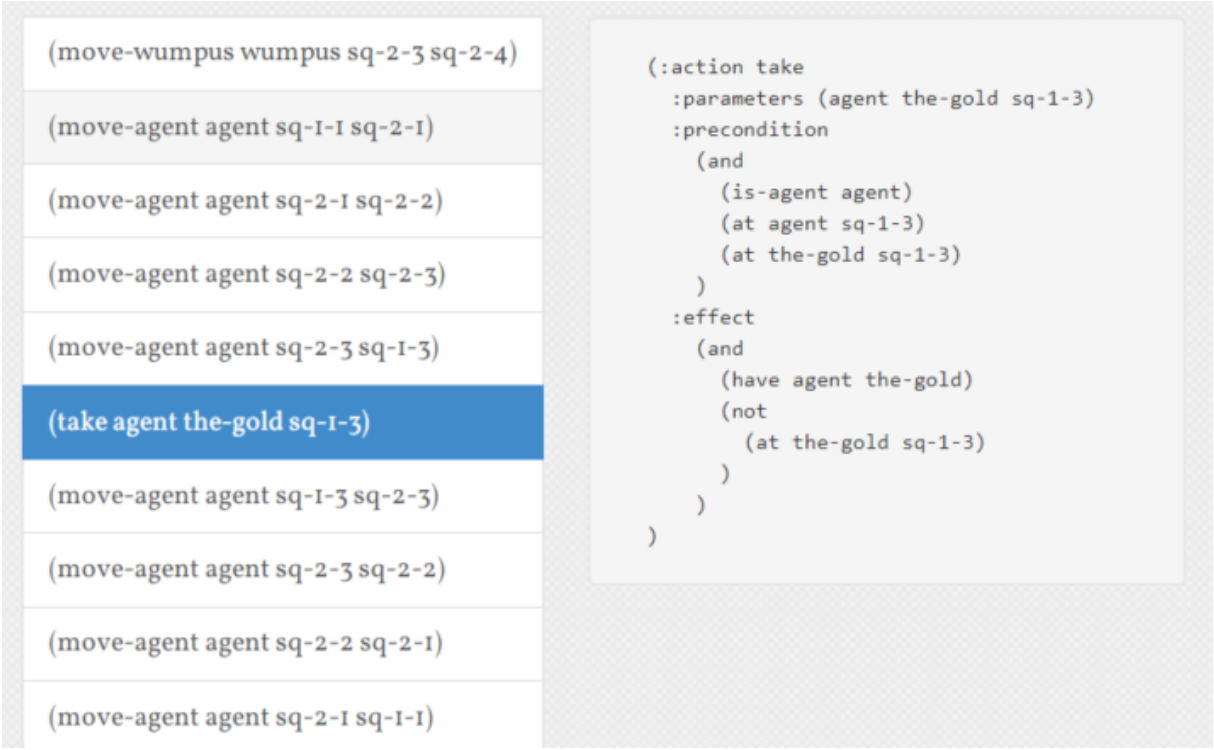
4)


(move-wumpus wumpus sq-2-3 sq-2-4)
(move-agent agent sq-1-1 sq-2-1)
(move-agent agent sq-2-1 sq-2-2)
(move-agent agent sq-2-2 sq-2-3)
(move-agent agent sq-2-3 sq-1-3)
(take agent the-gold sq-1-3)
(move-agent agent sq-1-3 sq-2-3)
(move-agent agent sq-2-3 sq-2-2)
(move-agent agent sq-2-2 sq-2-1)
(move-agent agent sq-2-1 sq-1-1)
(:action move-agent
:parameters (agent sq-2-2 sq-2-3)
:precondition
(and
(is-agent agent)
(at agent sq-2-2)
(adj sq-2-2 sq-2-3)
(not
(pit sq-2-3)
)
(not
(wumpus-in sq-2-3)
)
)
:effect
(and
(not
(at agent sq-2-2)
)
(at agent sq-2-3)
)
)

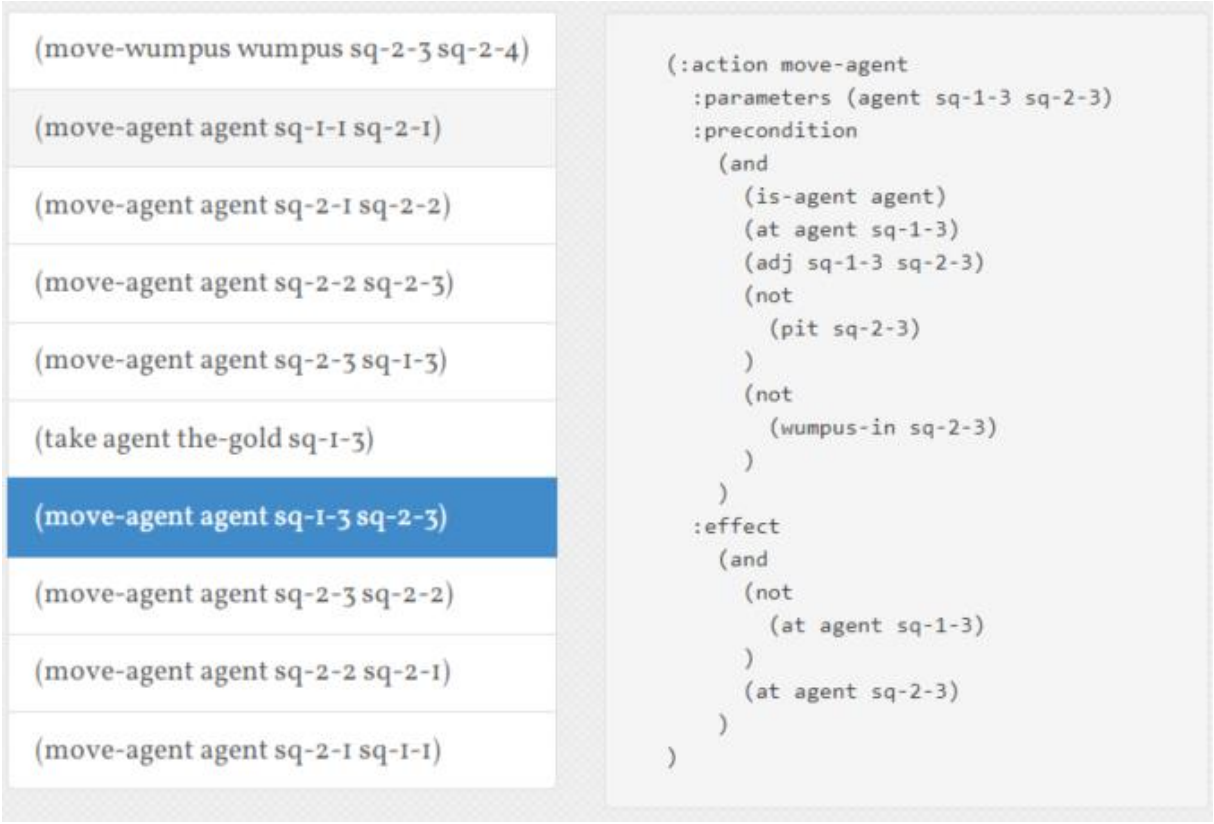
5)


(move-wumpus wumpus sq-2-3 sq-2-4)
(move-agent agent sq-1-1 sq-2-1)
(move-agent agent sq-2-1 sq-2-2)
(move-agent agent sq-2-2 sq-2-3)
(move-agent agent sq-2-3 sq-1-3)
(take agent the-gold sq-1-3)
(move-agent agent sq-1-3 sq-2-3)
(move-agent agent sq-2-3 sq-2-2)
(move-agent agent sq-2-2 sq-2-1)
(move-agent agent sq-2-1 sq-1-1)
(:action move-agent
:parameters (agent sq-2-3 sq-1-3)
:precondition
(and
(is-agent agent)
(at agent sq-2-3)
(adj sq-2-3 sq-1-3)
(not
(pit sq-1-3)
)
(not
(wumpus-in sq-1-3)
)
)
:effect
(and
(not
(at agent sq-2-3)
)
(at agent sq-1-3)
)
)

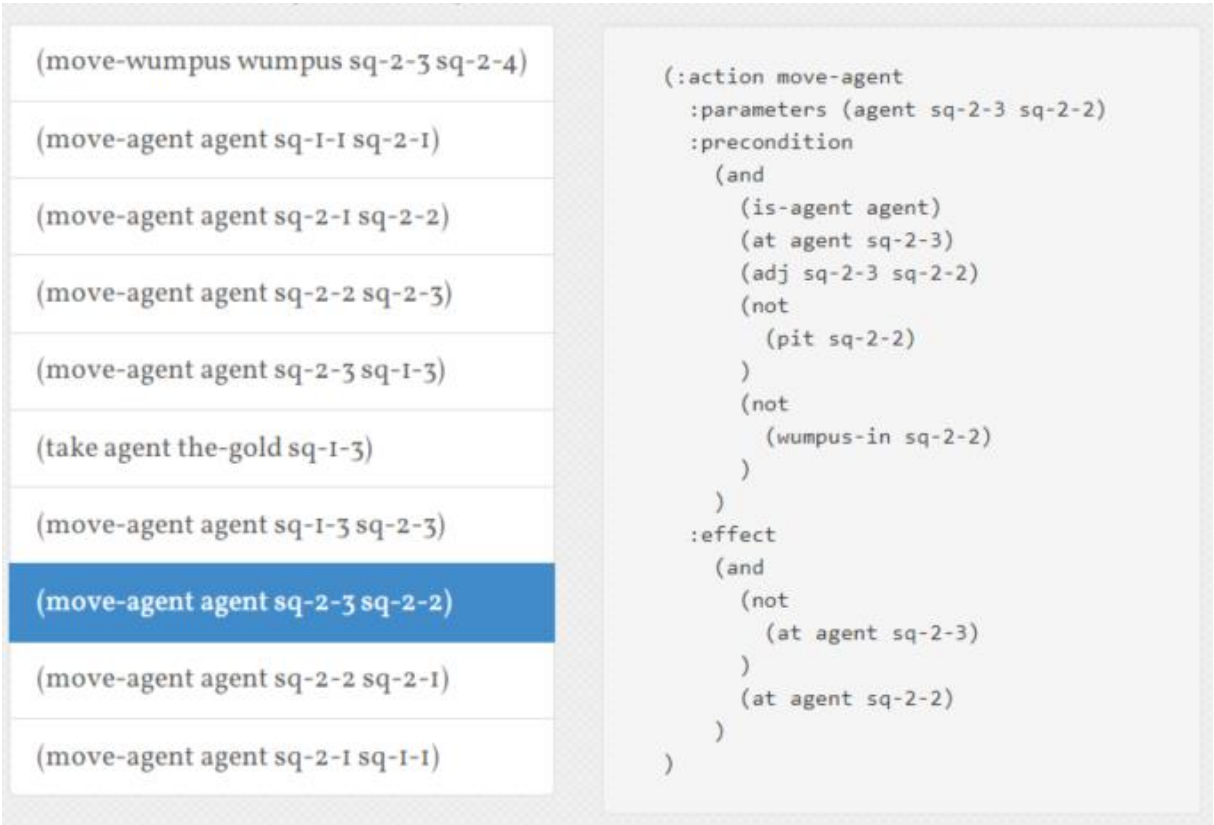
6)


(move-wumpus wumpus sq-2-3 sq-2-4)
(move-agent agent sq-1-1 sq-2-1)
(move-agent agent sq-2-1 sq-2-2)
(move-agent agent sq-2-2 sq-2-3)
(move-agent agent sq-2-3 sq-1-3)
(take agent the-gold sq-1-3)
(move-agent agent sq-1-3 sq-2-3)
(move-agent agent sq-2-3 sq-2-2)
(move-agent agent sq-2-2 sq-2-1)
(move-agent agent sq-2-1 sq-1-1)
(:action take
:parameters (agent the-gold sq-1-3)
:precondition
(and
(is-agent agent)
(at agent sq-1-3)
(at the-gold sq-1-3)
)
:effect
(and
(have agent the-gold)
(not
(at the-gold sq-1-3)
)
)
)

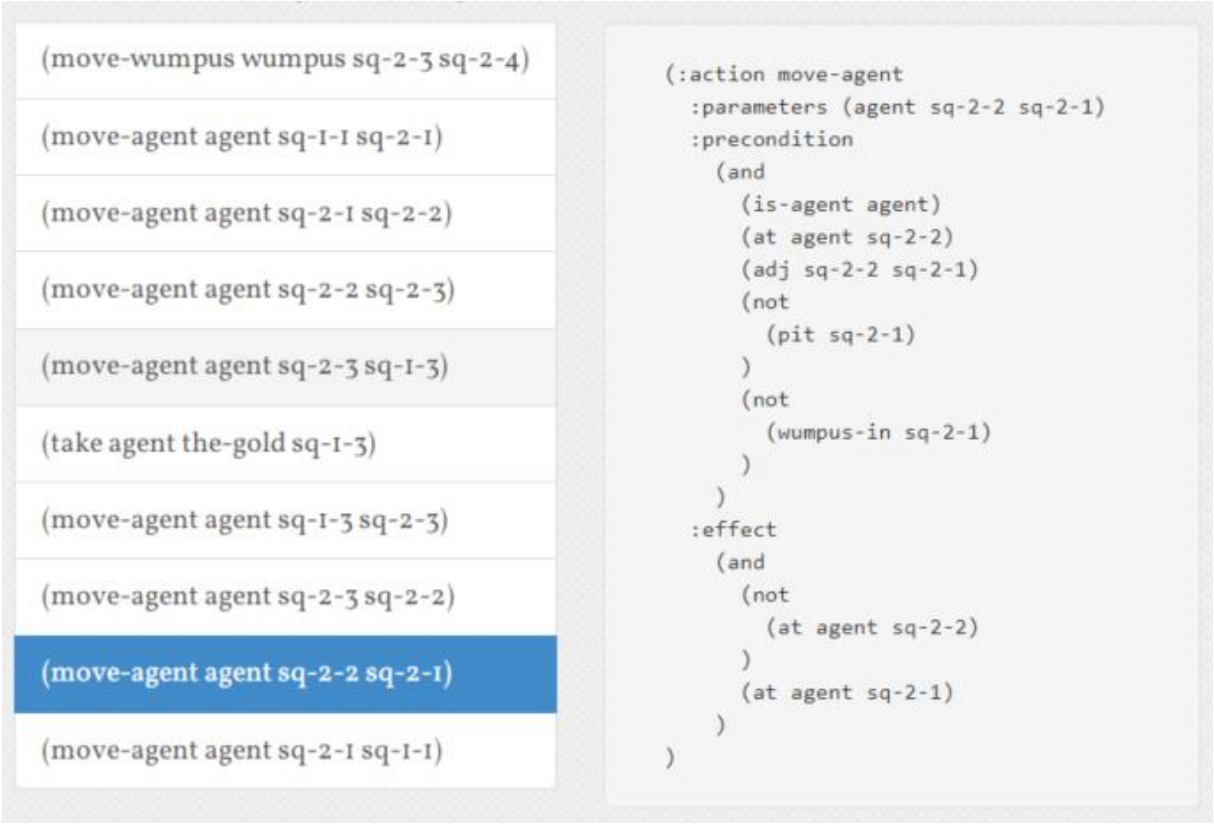
7)


(move-wumpus wumpus sq-2-3 sq-2-4)
(move-agent agent sq-1-1 sq-2-1)
(move-agent agent sq-2-1 sq-2-2)
(move-agent agent sq-2-2 sq-2-3)
(move-agent agent sq-2-3 sq-1-3)
(take agent the-gold sq-1-3)
(move-agent agent sq-1-3 sq-2-3)
(move-agent agent sq-2-3 sq-2-2)
(move-agent agent sq-2-2 sq-2-1)
(move-agent agent sq-2-1 sq-1-1)
(:action move-agent
:parameters (agent sq-1-3 sq-2-3)
:precondition
(and
(is-agent agent)
(at agent sq-1-3)
(adj sq-1-3 sq-2-3)
(not
(pit sq-2-3)
)
(not
(wumpus-in sq-2-3)
)
)
:effect
(and
(not
(at agent sq-1-3)
)
(at agent sq-2-3)
)
)

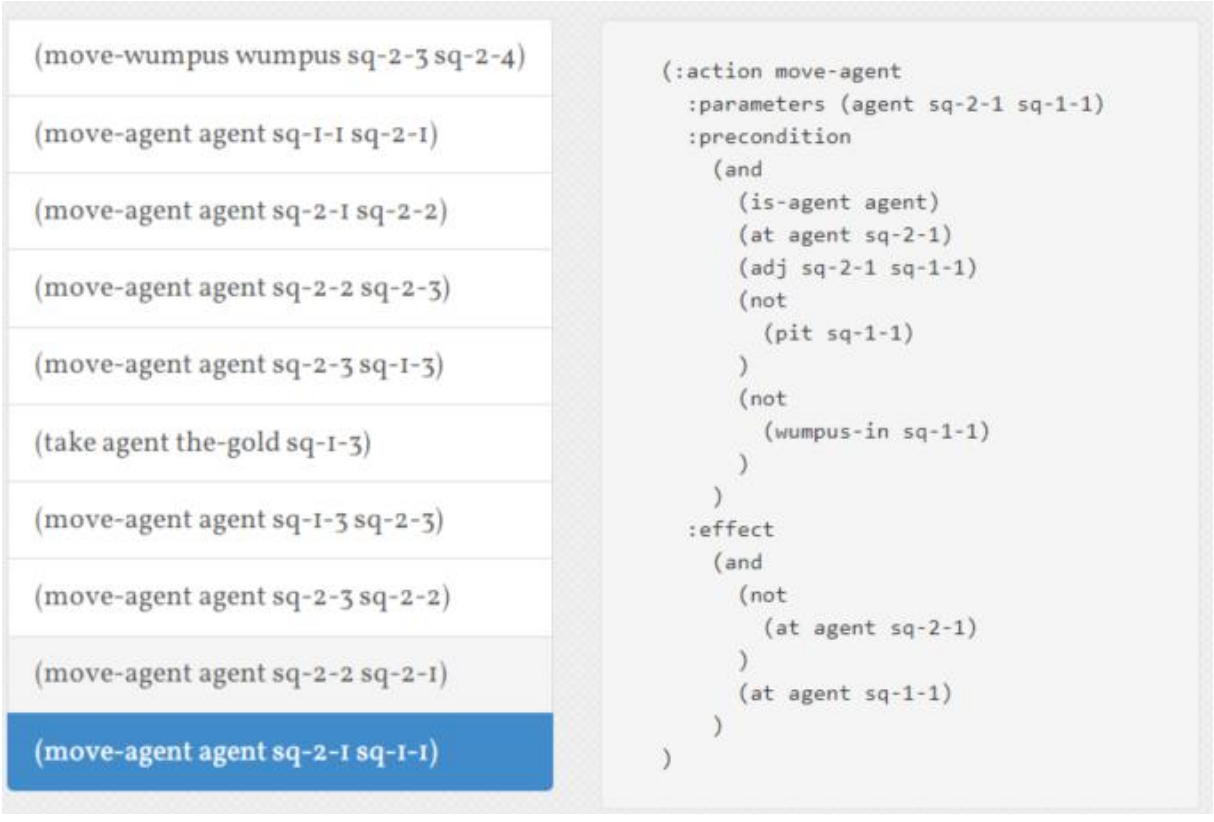
8)


(move-wumpus wumpus sq-2-3 sq-2-4)
(move-agent agent sq-1-1 sq-2-1)
(move-agent agent sq-2-1 sq-2-2)
(move-agent agent sq-2-2 sq-2-3)
(move-agent agent sq-2-3 sq-1-3)
(take agent the-gold sq-1-3)
(move-agent agent sq-1-3 sq-2-3)
(move-agent agent sq-2-3 sq-2-2)
(move-agent agent sq-2-2 sq-2-1)
(move-agent agent sq-2-1 sq-1-1)
(:action move-agent
:parameters (agent sq-2-3 sq-2-2)
:precondition
(and
(is-agent agent)
(at agent sq-2-3)
(adj sq-2-3 sq-2-2)
(not
(pit sq-2-2)
)
(not
(wumpus-in sq-2-2)
)
)
:effect
(and
(not
(at agent sq-2-3)
)
(at agent sq-2-2)
)
)

9)


(move-wumpus wumpus sq-2-3 sq-2-4)
(move-agent agent sq-1-1 sq-2-1)
(move-agent agent sq-2-1 sq-2-2)
(move-agent agent sq-2-2 sq-2-3)
(move-agent agent sq-2-3 sq-1-3)
(take agent the-gold sq-1-3)
(move-agent agent sq-1-3 sq-2-3)
(move-agent agent sq-2-3 sq-2-2)
(move-agent agent sq-2-2 sq-2-1)
(move-agent agent sq-2-1 sq-1-1)
(:action move-agent
:parameters (agent sq-2-2 sq-2-1)
:precondition
(and
(is-agent agent)
(at agent sq-2-2)
(adj sq-2-2 sq-2-1)
(not
(pit sq-2-1)
)
(not
(wumpus-in sq-2-1)
)
)
:effect
(and
(not
(at agent sq-2-2)
)
(at agent sq-2-1)
)
)

10)


(move-wumpus wumpus sq-2-3 sq-2-4)
(move-agent agent sq-1-1 sq-2-1)
(move-agent agent sq-2-1 sq-2-2)
(move-agent agent sq-2-2 sq-2-3)
(move-agent agent sq-2-3 sq-1-3)
(take agent the-gold sq-1-3)
(move-agent agent sq-1-3 sq-2-3)
(move-agent agent sq-2-3 sq-2-2)
(move-agent agent sq-2-2 sq-2-1)
(move-agent agent sq-2-1 sq-1-1)
(:action move-agent
:parameters (agent sq-2-1 sq-1-1)
:precondition
(and
(is-agent agent)
(at agent sq-2-1)
(adj sq-2-1 sq-1-1)
(not
(pit sq-1-1)
)
(not
(wumpus-in sq-1-1)
)
)
:effect
(and
(not
(at agent sq-2-1)
)
(at agent sq-1-1)
)
)