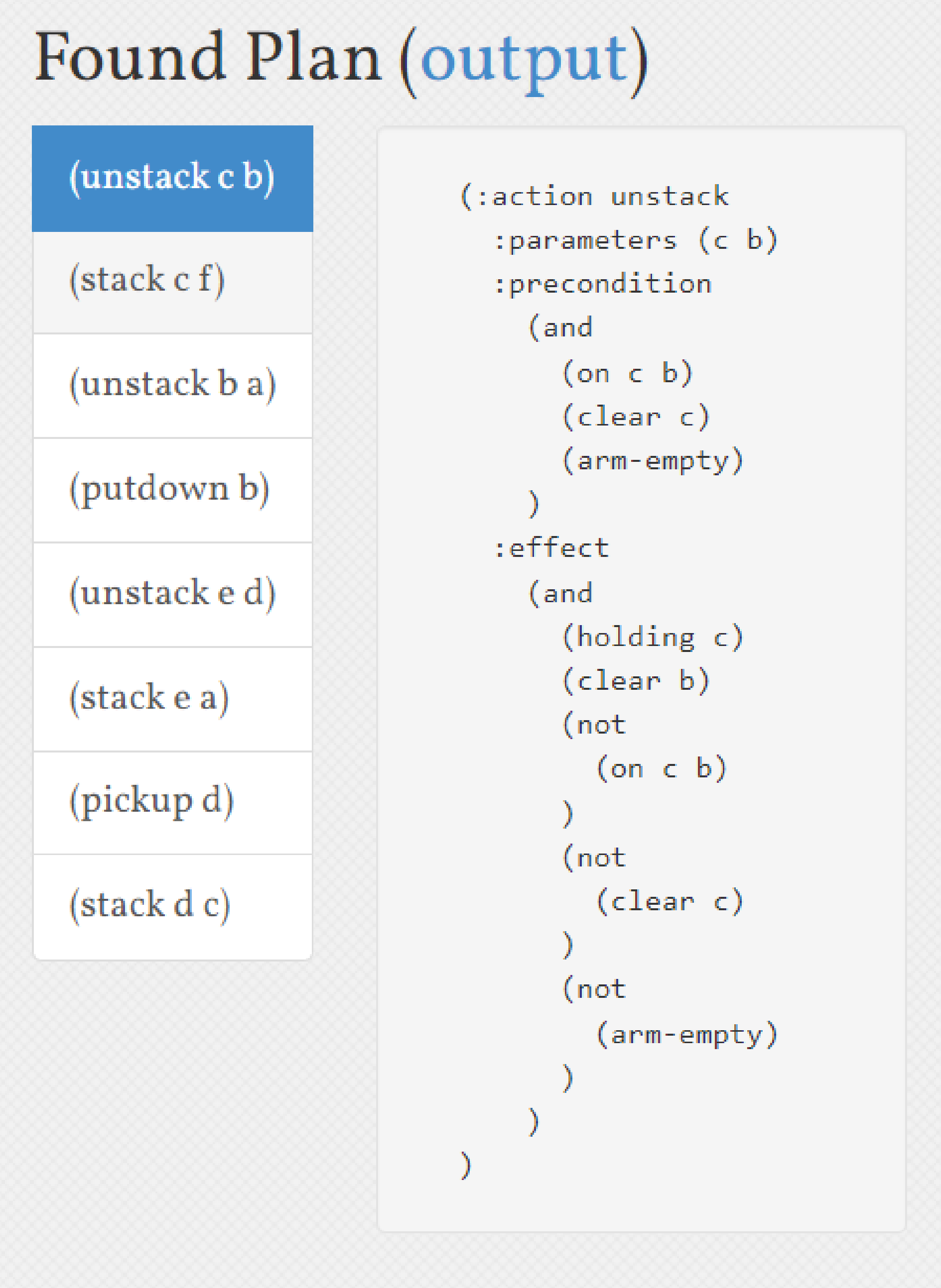
B877006 김윤성 pddl screenshot

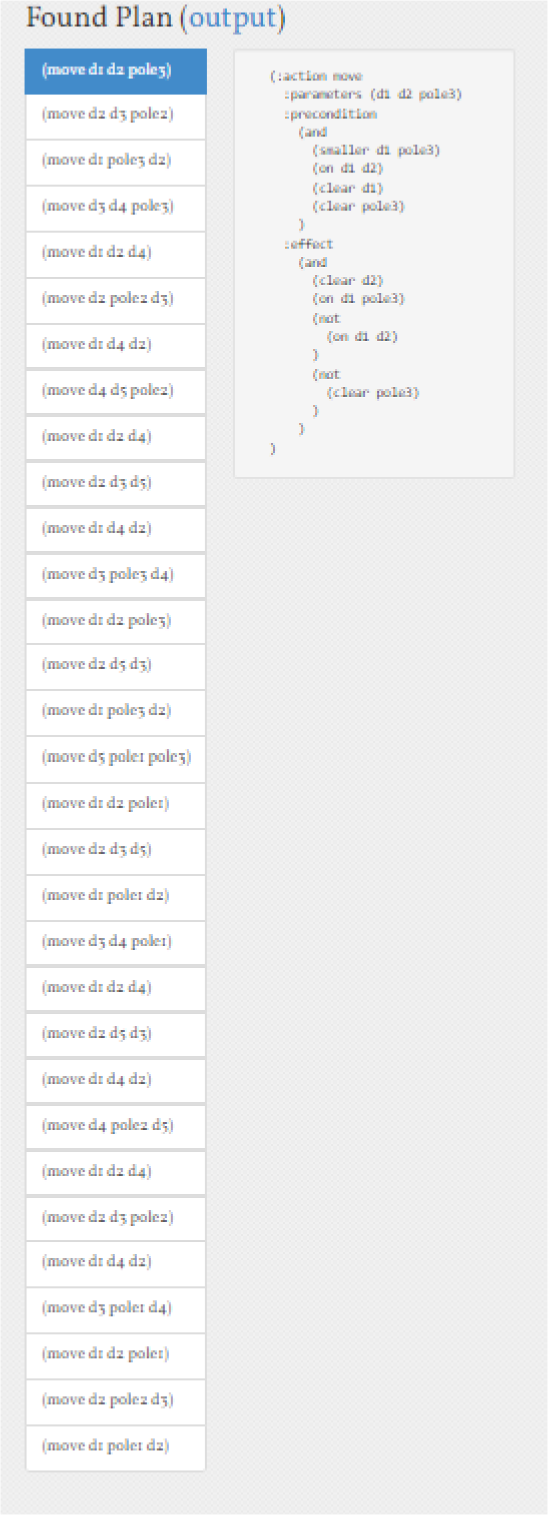
1.



Found Plan (output)

```
(:action unstack
  :parameters (c b)
  :precondition
    (and
      (on c b)
      (clear c)
      (arm-empty)
    )
  :effect
    (and
      (holding c)
      (clear b)
      (not
        (on c b)
      )
      (not
        (clear c)
      )
      (not
        (arm-empty)
      )
    )
)
```

2.

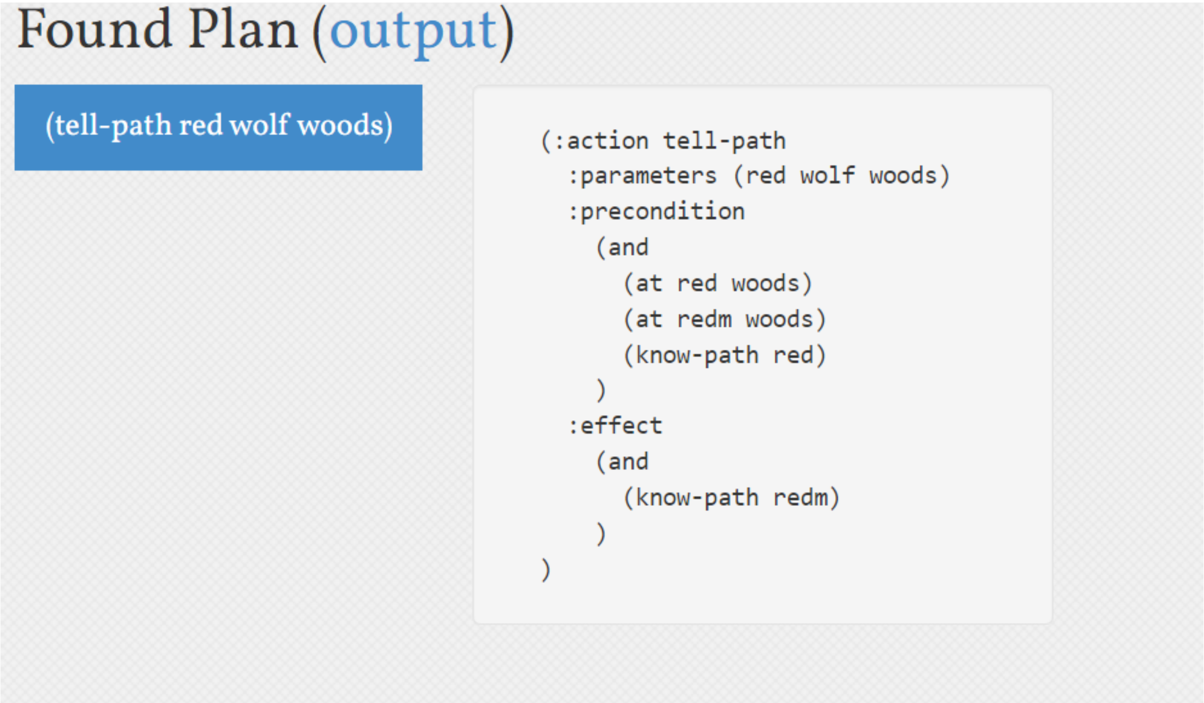


Found Plan (output)

```
(move d1 d2 pole3)
(move d2 d3 pole2)
(move d1 pole3 d2)
(move d3 d4 pole3)
(move d1 d2 d4)
(move d2 pole2 d3)
(move d1 d4 d2)
(move d4 d5 pole2)
(move d1 d2 d4)
(move d2 d3 d5)
(move d1 d4 d2)
(move d3 pole3 d4)
(move d1 d2 pole3)
(move d2 d5 d3)
(move d1 pole3 d2)
(move d5 pole1 pole3)
(move d1 d2 pole1)
(move d2 d3 d5)
(move d1 pole1 d2)
(move d3 d4 pole1)
(move d1 d2 d4)
(move d2 d5 d3)
(move d1 d4 d2)
(move d4 pole2 d5)
(move d1 d2 d4)
(move d2 d3 pole2)
(move d1 d4 d2)
(move d3 pole1 d4)
(move d1 d2 pole1)
(move d2 pole2 d3)
(move d1 pole1 d2)
```

```
(:action move
  :parameters (d1 d2 pole3)
  :precondition
    (and
      (smaller d1 pole3)
      (on d1 d2)
      (clear d1)
      (clear pole3)
    )
  :effect
    (and
      (clear d2)
      (on d1 pole3)
      (not
        (on d1 d2)
      )
      (not
        (clear pole3)
      )
    )
)
```

3-1

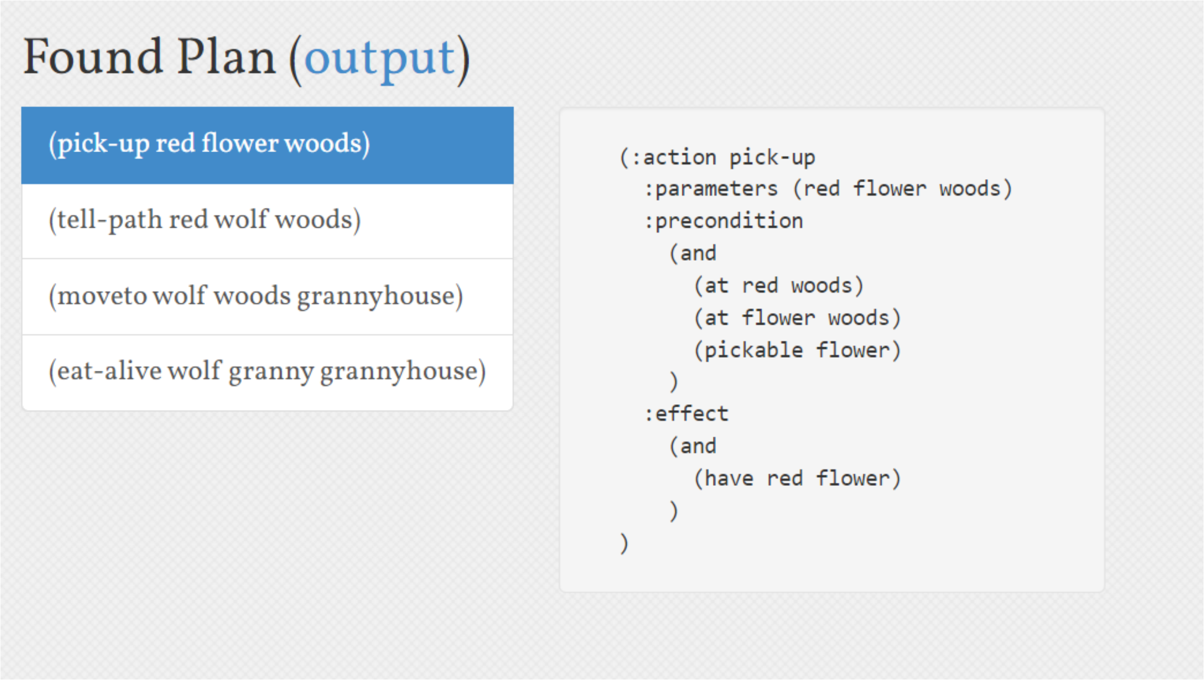


Found Plan (output)

(tell-path red wolf woods)

```
(:action tell-path
  :parameters (red wolf woods)
  :precondition
    (and
      (at red woods)
      (at redm woods)
      (know-path red)
    )
  :effect
    (and
      (know-path redm)
    )
)
```

3-2,3



Found Plan (output)

(pick-up red flower woods)

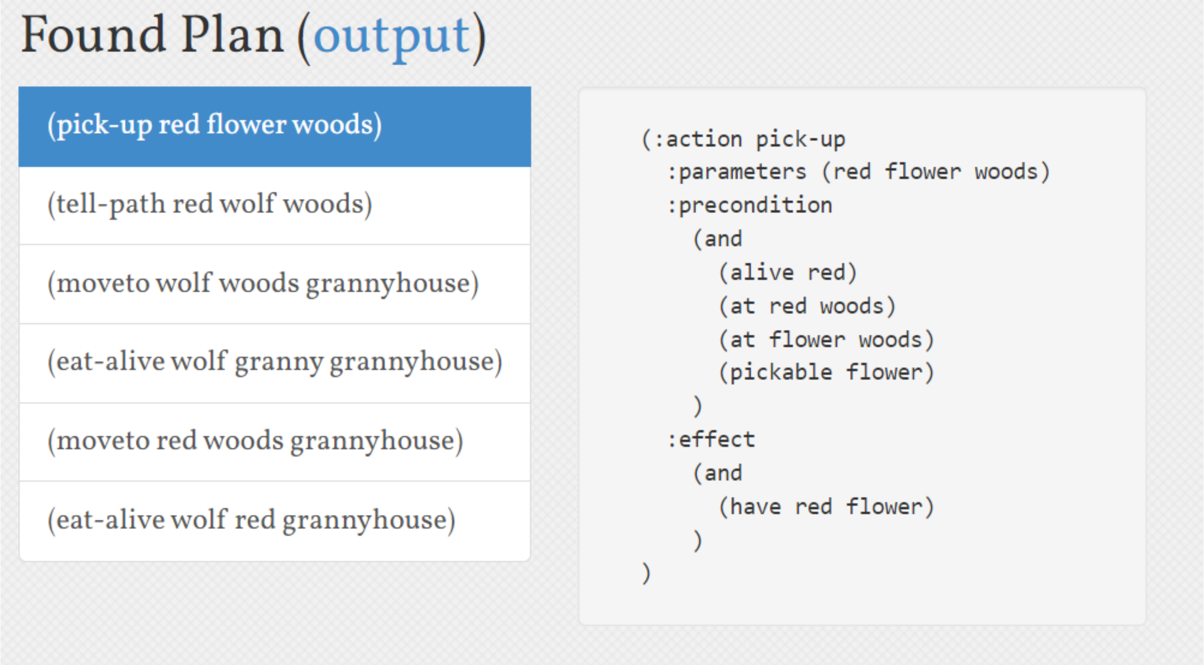
(tell-path red wolf woods)

(moveto wolf woods grannyhouse)

(eat-alive wolf granny grannyhouse)

```
(:action pick-up
  :parameters (red flower woods)
  :precondition
    (and
      (at red woods)
      (at flower woods)
      (pickable flower)
    )
  :effect
    (and
      (have red flower)
    )
)
```

3-4

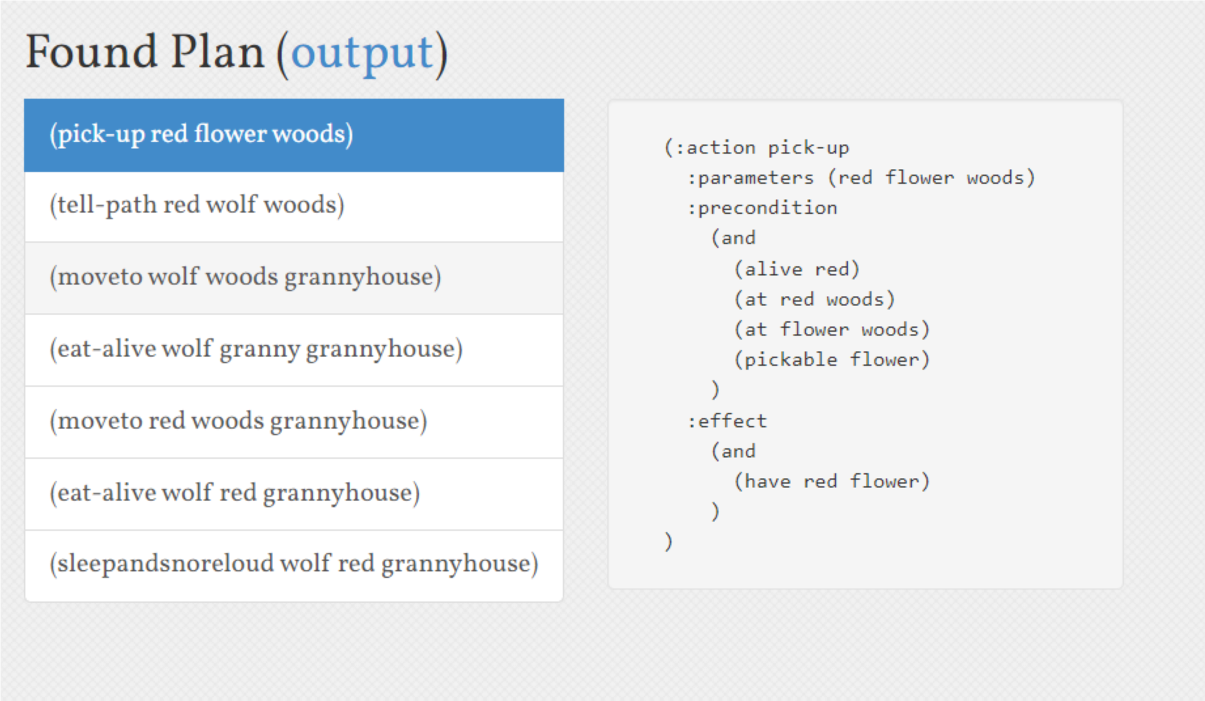


Found Plan (output)

- (pick-up red flower woods)
- (tell-path red wolf woods)
- (moveto wolf woods grannyhouse)
- (eat-alive wolf granny grannyhouse)
- (moveto red woods grannyhouse)
- (eat-alive wolf red grannyhouse)

```
(:action pick-up
  :parameters (red flower woods)
  :precondition
    (and
      (alive red)
      (at red woods)
      (at flower woods)
      (pickable flower)
    )
  :effect
    (and
      (have red flower)
    )
)
```

3-5

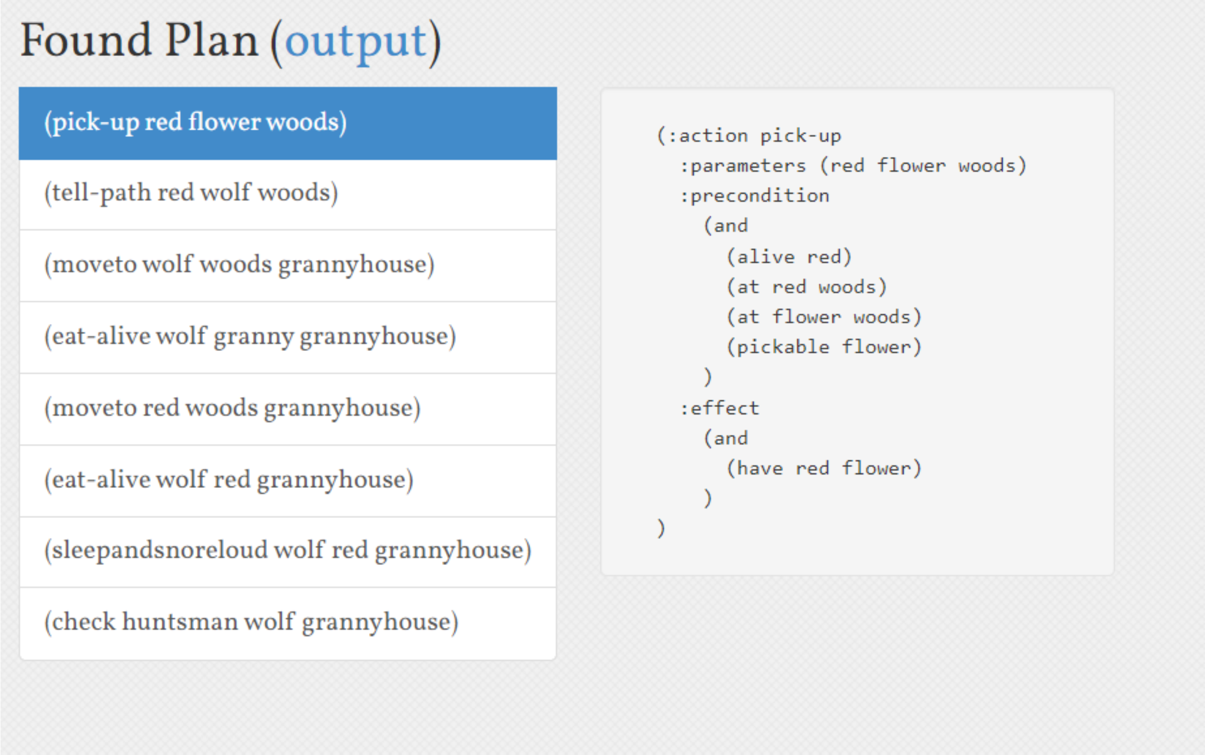


Found Plan (output)

- (pick-up red flower woods)
- (tell-path red wolf woods)
- (moveto wolf woods grannyhouse)
- (eat-alive wolf granny grannyhouse)
- (moveto red woods grannyhouse)
- (eat-alive wolf red grannyhouse)
- (sleepandsnoreloud wolf red grannyhouse)

```
(:action pick-up
  :parameters (red flower woods)
  :precondition
    (and
      (alive red)
      (at red woods)
      (at flower woods)
      (pickable flower)
    )
  :effect
    (and
      (have red flower)
    )
)
```

3-6



Found Plan (output)

(pick-up red flower woods)

(tell-path red wolf woods)

(moveto wolf woods grannyhouse)

(eat-alive wolf granny grannyhouse)

(moveto red woods grannyhouse)

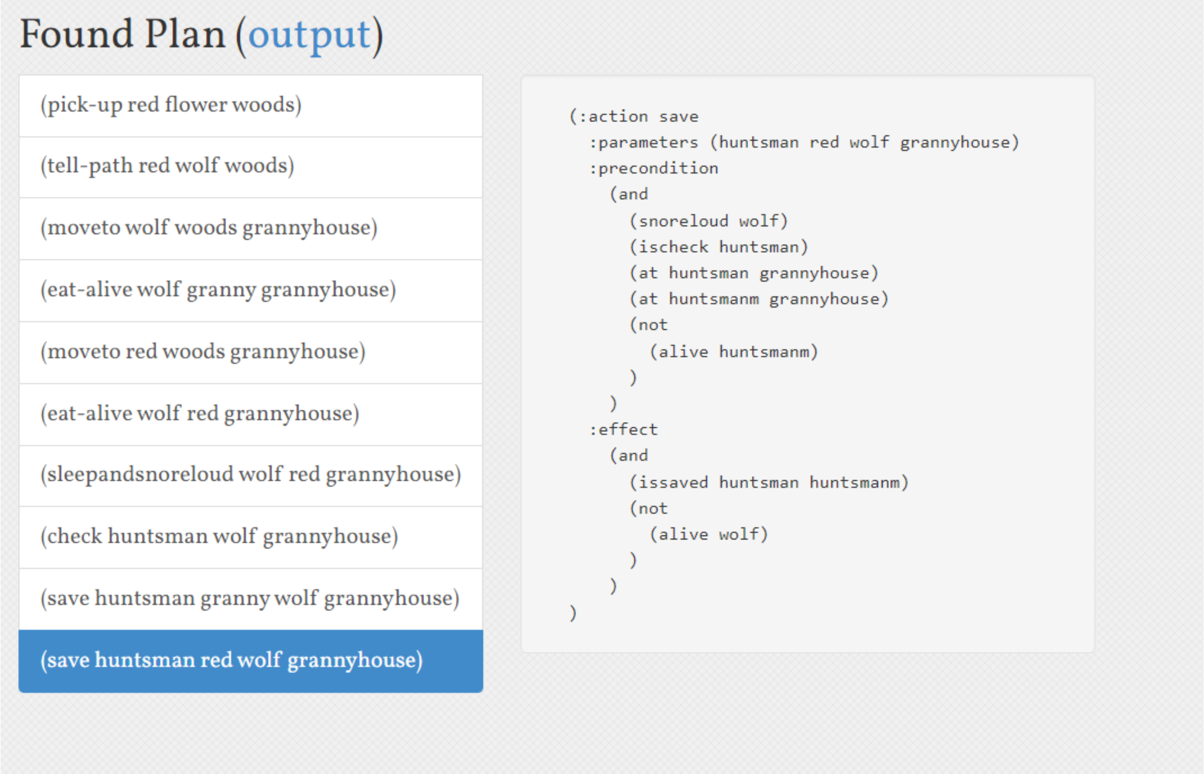
(eat-alive wolf red grannyhouse)

(sleepandsnoreloud wolf red grannyhouse)

(check huntsman wolf grannyhouse)

```
(:action pick-up
  :parameters (red flower woods)
  :precondition
    (and
      (alive red)
      (at red woods)
      (at flower woods)
      (pickable flower)
    )
  :effect
    (and
      (have red flower)
    )
)
```

3-7



Found Plan (output)

(pick-up red flower woods)

(tell-path red wolf woods)

(moveto wolf woods grannyhouse)

(eat-alive wolf granny grannyhouse)

(moveto red woods grannyhouse)

(eat-alive wolf red grannyhouse)

(sleepandsnoreloud wolf red grannyhouse)

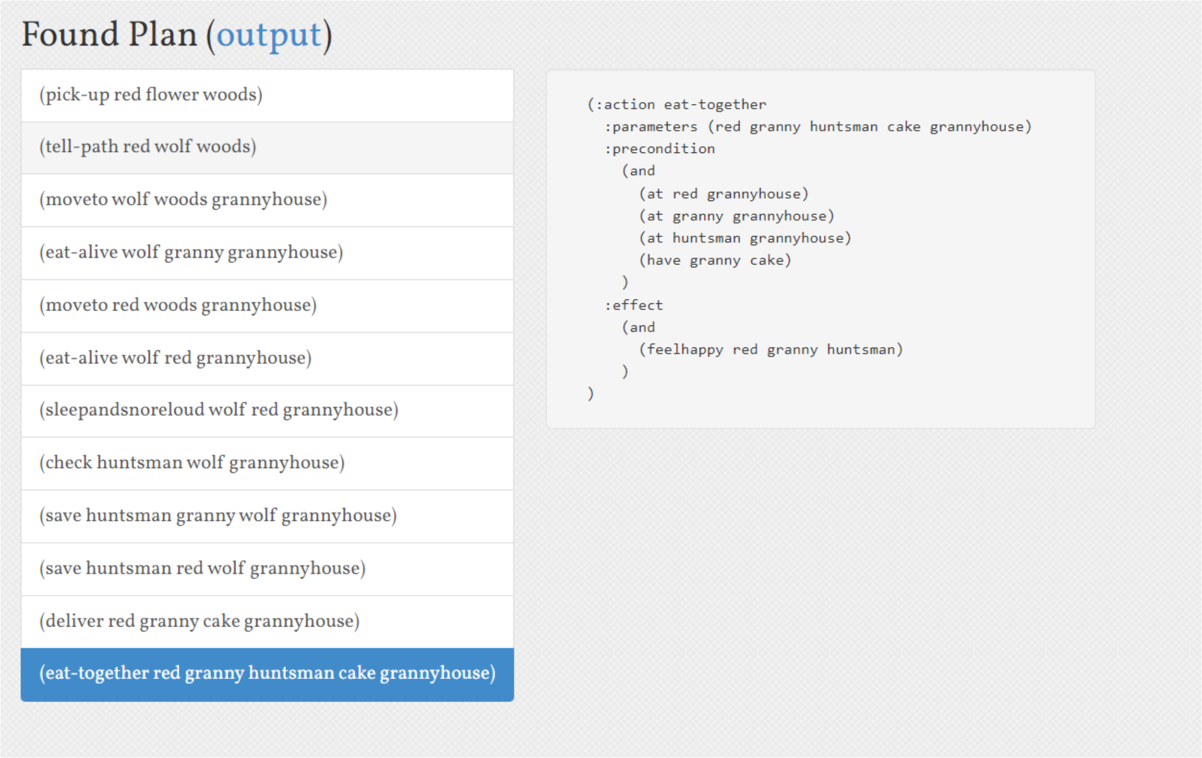
(check huntsman wolf grannyhouse)

(save huntsman granny wolf grannyhouse)

(save huntsman red wolf grannyhouse)

```
(:action save
  :parameters (huntsman red wolf grannyhouse)
  :precondition
    (and
      (snoreloud wolf)
      (ischeck huntsman)
      (at huntsman grannyhouse)
      (at huntsmanm grannyhouse)
      (not
        (alive huntsmanm)
      )
    )
  :effect
    (and
      (issaved huntsman huntsmanm)
      (not
        (alive wolf)
      )
    )
)
```

3-8



Found Plan (output)

(pick-up red flower woods)

(tell-path red wolf woods)

(moveto wolf woods grannyhouse)

(eat-alive wolf granny grannyhouse)

(moveto red woods grannyhouse)

(eat-alive wolf red grannyhouse)

(sleepandsnoreloud wolf red grannyhouse)

(check huntsman wolf grannyhouse)

(save huntsman granny wolf grannyhouse)

(save huntsman red wolf grannyhouse)

(deliver red granny cake grannyhouse)

(eat-together red granny huntsman cake grannyhouse)

```
(:action eat-together
  :parameters (red granny huntsman cake grannyhouse)
  :precondition
    (and
      (at red grannyhouse)
      (at granny grannyhouse)
      (at huntsman grannyhouse)
      (have granny cake)
    )
  :effect
    (and
      (feelhappy red granny huntsman)
    )
)
```