CMPE480 Project 2

Elif Çalışkan

2016400183

Predicates:

(atShore ?obj - cargo ?sh - shore): true if cargo obj is at shore sh

(eats ?obj1 - cargo ?obj2 - cargo): true if cargo obj1 eats cargo obj2

(on ?sh - shore): true if farmer is on shore sh

(onBoat ?obj): true if cargo obj is at shore sh

Actions:

TAKE_TO_BOAT: if cargo obj is at shore sh, the farmer is on sh and there is no pair of cargos that eats one another; with the help of this action, the farmer is not on sh, obj is not at shore sh and obj is on boat anymore.

TAKE_TO_SHORE: if cargo obj is not at shore sh, the farmer is on boat; with the help of this action, the farmer is on sh, obj is at sh and obj is not on boat anymore.

GO_TO_SHORE: if the farmer is on sh1 and there is no danger of cargos eating one another at shore sh1, farmer goes to sh2 from now on.

Objects

wolf - cargo

goat - cargo

cabbage - cargo

shore1 - shore

shore2 - shore

Initialization

(atShore wolf shore1) (atShore goat shore1) (atShore cabbage shore1)

(eats wolf goat) (eats goat cabbage) (on shore1)

At first; wolf, goat and cabbage is at shore1. Wolf eats goat if they are alone, goat eats cabbage if they are alone. Farmer is on shore1.

Goal

(and (atShore wolf shore2) (atShore goat shore2) (atShore cabbage shore2) (on shore2))

We want to bring every cargo to shore2.

Found Plan:

Found Plan (output)

```
(take_to_boat goat shore1)
(take_to_shore goat shore2)
(go_to_shore shore2 shore1)
(take_to_boat wolf shore1)
(take_to_shore wolf shore2)
(take_to_boat goat shore2)
(take_to_shore goat shore1)
(take_to_boat cabbage shore1)
(take_to_shore cabbage shore2)
(go_to_shore shore2 shore1)
(take_to_boat goat shore1)
(take_to_shore goat shore2)
```