List 2[1]

Exercise 1

Consider the following planning task:

- You are trapped in the cellar of a building with a switch board full of light switches. In the rooms above you there is a vampire. Luckily, there also is a vampire slayer in those rooms. To keep things simple, we consider only room layouts that are circular corridors where each room has a clockwise and an anti-clockwise neighbor.
- The vampire avoids the light (yes, even if it is only artificial light). Whenever the light in the vampire's room is switched on, it moves to a neighboring room. If one of the rooms is dark, it will move there, preferring the anti-clockwise one if both are dark. If both neighboring rooms are bright, it will move clockwise.
- The slayer tries to stay in the light. If the light in her room is switched off, she moves to a neighboring room. She moves clockwise if that room is bright and anti-clockwise otherwise.
- If the two of them meet in a room they will fight. The vampire wins the fight in a dark room unless there is garlic in that room. In bright rooms or in rooms with garlic, the slayer wins.
- All you can do is use the switch board to toggle lights and watch the fight, when it happens.

(a) There is a partial model of this domain in the directory vampire. Complete it by adding the effects of `toggle-light` and `watch-fight`. Do not add new actions or predicates.

The directory also contains instances which you can use for debugging. Their optimal plan costs are 6, 4, 7, 5, 4, 12, 11, 10, 13, and 8. Use INVAL for debugging. You can find it in the directory INVAL.

(b) PDDL uses first-order predicate logic to model planning tasks. However, the models discussed in the lecture are all based on propositional logic. Most planners convert PDDL into one of the propositional models in a step called *grounding*. The directory `preprocess` contains a Python tool to do this step. The call

```
./preprocess/ground.py vampire/domain.pddl vampire/p01.pddl
```

will create a new domain file `vampire/domain_grounded_for_p01.pddl` and a new task file `vampire/p01_grounded.pddl`. Repeat this for all task files and describe the effect of the grounding procedure.

[1] Exercício de Malte Helmert.

Exercise 2

Look up the top 5 planners in the sequential optimal and sequential satisficing track of the International Planning Competition (IPC) 2014. For each of them, list their search direction, search state representation, search algorithm, heuristic, and other interesting aspects analogously to the slides "*12. Planning Formalisms*". For planners that use a portfolio of other planners it is sufficient to list the names of the other planners.