Assignment2

Q1.

A **goal-based agent** has the same properties of a model-based agent plus some goal information to describe desirable situations to aim for. This further information, called goals, get added to the internal model replacing the condition-action rules so that the agent can make better choices on the action to perform. In this way its knowledge is more flexible and can be modified in time, making the whole model more complex, though.

A **utility-based agent** has similar properties of a goal-based agent, but its decisions are driven by a utility function instead of the goals. This type of agents makes decisions trying to maximize their utility function, which takes into consideration more information, preferences and aspects compared to the goals.

The main difference is then that while goal-based agents are focused on achieving set objectives, utility-based agents pick actions based on how well they improve their overall benefit or utility.

Q2.

Let's analyze all environment properties and the assumptions that would make them valid in this scenario.

- **Fully observable**: the agent's sensors give it access to the entire state of the environment. The agent is equipped with cameras that have a view range that covers the whole field and a GPS sensor that tells it which square it is in.
- **Partially observable**: the agent's sensors do not give it access to a complete state of the environment. The agent's equipped cameras are not able to cover the entire field, meaning that the agent can't tell whether or not there is a star in a square far away from its current position.
- **Deterministic**: the agent's moving and picking actuators are always correctly working as well as its sensors (camera and GPS, for instance). In this case there are no stars randomly appearing on the field, so that the agent can predict the next environment state given the current status and the action performed.
- **Stochastic**: the agent's sensors do not give it access to a complete state of the environment or the actuators are not completely reliable and sometimes they malfunction. In these cases, the next environment state is not fully determined by the current one and the performed action. For instance, the picking actuator could successfully pick up a star only in 80% of cases, while in the remaining 20% the star stays on the square.
- **Dynamic**: the environment itself changes as time passes, so the agent has to keep observing the whole environment to detect the changes. For instance, the stars could keep appearing and disappearing from the squares as time goes on.
- **Static**: the environment does not change on its own, but only when the agent performs some actions. In this case, stars could be placed on the field once and then they can only be picked up by the agent.
- **Sequential**: the current decision taken by the agent affects the future decisions. The agent's goal could be to pick up the stars by minimizing the travelled distance, so each decision has an effect on future decisions since the best path choice is a "global" choice over the whole field.

- **Episodic**: each decision taken by the agent given a percept is independent of the previously taken decisions. The agent's goal is just to pick up all the stars, without any additional constraint and one episode corresponds to picking up a star: the decision to pick up a star is independent of the others, since there is no need for the agent to consider the overall state of the grid. In this case the environment does not require the agent to remember past states or plan for future actions based on previous experience.
- **Known**: the outcomes for all actions are given. The agent's knowledge includes the correct behavior of all actuators (correctly calibrated) and sensors (accurate measurements).
- **Unknown**: the outcomes for the agent's actions are not completely given. The agent has a partial knowledge of the behavior of the actuators (not calibrated) and sensors (not accurate). These issues do not allow the agent to move in a known environment, but it could get to such environment through learning, by observing the behavior of actuators and sensors and the effects of its actions to over time.
- **Continuous**: the location (and speed if we also assume a speedometer equipped on the robot) of the agent can smoothly range over time through many continuous values. This holds if we assume that the robot can move up, down, right and left by an arbitrary number of meters.
- **Discrete**: the agent's location can only range through a fixed number of squares, if we assume that the robot can move up, down, right and left only by one square at a time. Also, the picking action is atomic and the robot has no sensors that can detect the exact current position (and angle) of its arm, which is then always considered either standing by or fully extended picking up a star.
- **Single-agent**: the robot is the only agent involved in this environment. The stars, the only other entities involved, just stay still since they can only be picked up by the agent.
- **Multi-agent**: the robot and the stars are all agents involved in this environment. The agent's goal is to pick up as many stars as possible, while the stars' one is to not get picked up by the robot. In this scenario, both the robot and the stars try to maximize their own performance measure that depends on the robot's behavior.