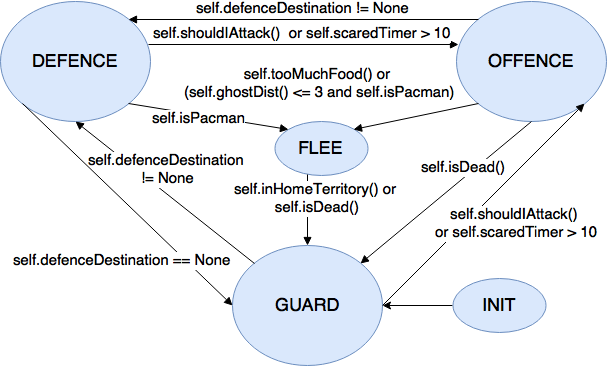
**Pacman Capture The Flag**

**Agent design:**

****

Our agent has four possible behavior states, Flee, Defence, Offence and Guard. For each move, the agent updates its behavior state as outlined in Figure 1. The approach for each of the four behavior states is as follows:

**Guard:** The agents seek the center position at the top half and bottom half at the board, respectively. A greedy best first search with *mazeDistance()* as the heuristic is done every move to decide on the action that makes the agent move towards the desired position.

**Defence:** The agent moves towards *self.defenceDestination* using a greedy best first search with *mazeDistance()* as the heuristic. Actions that take the pacman out of its home territory are not allowed, and STOP is only allowed if the agent has no other option.

**Offence:** Monte Carlo searches for the expanding trees from the successors of the current position are taken. Each search returns a sum of weighted features at the end position of the expanded tree. The action taken is the one that leads to the successor with the best sum of Monte Carlo scores. If a successor of the current position contains food, Monte Carlo searches are not performed for that successor. A piece of food with a distance of 1 is then weighted highly even if trees expanding from that successor do not generate a good Monte Carlo score.If the minimum distance to the closest piece of food is large, the Monte Carlo search results does not correspond with our desired behavior, and the agent does a greedy best first search for the closest piece of food.

**Flee:** The agent does a blind breadth first search for the shortest path to its home territory that does not contain a ghost. The agent takes the first action on the returned path of action, and the search is redone for each turn.

**Challenges and possible improvements:**

Our design is heavily based on decision trees, which makes it difficult to make an agent that performs well on all possible layouts and opponents. A higher level of automation could increase the generalization of the agent and make it more robust for a wide range of challenges.

Tuning the weighting of the parameters for the features evaluated by the Monte Carlo searches have been challenging, causing the agent to oscillate or take strange moves in some situations. A possible improvement to avoid the agent of getting stuck in an oscillating position could be to evaluate the distance to the center of the opponent’s part of the board.

The design of the state machine only allows the agent to enter Offence behavior state if an opponent is killed or it is scared. T

**Performance of implemented approaches:**

The behavior states and the switching between them are working quite well. For each move, the agent searches for observable opponents and positions where food is eaten. If less than two opponents are observed and the position where the food is eaten is not within a distance of 1 from the closest observed opponent, the agent detect that there is an opponent at this position.

Detected positions are assigned to one of the agents on the team, and are saved in *self.defenceDestination*, causing the agent to enter Defence behavior state and chase the position until it is reached or no longer detectable.

*self.shouldIAttack()* checks if an opponent are killed, and the agent with the greatest distance to any detectable opponent then enters Offence behavior state. This makes our agent quite cautious, as it is not allowed to enter the opponent’s part of the board unless it is scared and therefore useless as a defender, or an opponent is killed and therefore is a weaker defender for the next moves.

* A clear written description of the design decisions made, approaches taken, challenges experienced, and possible improvements [3 marks]
* An experimental section that justifies and explains the performance of the approaches implemented [2 marks]
* a report.pdf of 2 pages, explaining your agents and doing experiments to show how different variations of your agent perform against each other.
* In your report.pdf, you must describe the techniques that you have implemented in your Agent for the tournament. Please clearly state any assumptions that you make. Include an analysis of the strengths and weaknesses of your techniques.