Explanation of the chosen heuristic:

1. **For trickySearch:**
   - Without heuristic: Path found with a total cost of **60**, nodes expanded: **16,688**, time: **57.6 seconds**.
   - With heuristic: Path found with a total cost of **60**, nodes expanded: **6,126**, time: **2.9 seconds**.

   The **costs** in both cases are the same (60), meaning the heuristic did not overestimate the cost.

2. **For tinySearch:**
   - Without heuristic: Path found with a total cost of **27**, nodes expanded: **5,057**, time: **6.1 seconds**.
   - With heuristic: Path found with a total cost of **27**, nodes expanded: **423**, time: **0.0 seconds**.

   Again, the **costs** are the same (27) in both cases, so the heuristic did not overestimate the cost.

3. **For testSearch:**
   - Without heuristic: Path found with a total cost of **7**, nodes expanded: **14**, time: **0.0 seconds**.
   - With heuristic: Path found with a total cost of **7**, nodes expanded: **12**, time: **0.0 seconds**.

   The **costs** are again identical (7).

In all three cases (trickySearch, tinySearch, and testSearch), the **path cost remains the same** whether or not a heuristic is used. This means that the heuristic **does not overestimate** the actual cost and is therefore **admissible**. In the heuristic, I am using Manhattan distances to calculate the distance from Pacman's current position to the nearest food. The Manhattan distance satisfies the triangle inequality, meaning that for any three points AAA, BBB, and CCC, the direct distance from AAA to CCC is always less than or equal to the distance from AAA to BBB plus the distance from BBB to CCC. This ensures that the heuristic does not overestimate the cost to the goal. Admissible heuristics also tend to be consistent as well.