

Design Challenge 1: Square Dance

The design challenge today will be a good old robot square dance! Your goal is to make your robot move in a square (please keep the side lengths between 50 and 75 cm). We will judge you based on the scoring guide at the bottom of this page. Use what you learned in the previous activities and apply it to this design challenge.

1 Strategize

Before we begin writing code, we need to compute the times required to move in a square.

1. Given a translational velocity, (t_v), of 100 mm/s , find the time needed to drive 60 cm .
2. Given a rotational velocity, (r_v), of 1571 mrad/s , compute the time needed to rotate 90 degrees . Remember, $90 \text{ degrees} = \pi/2 \text{ radians}$, and $\pi \text{ radians} \approx 3.142 = 3142 \text{ milliradians}$.

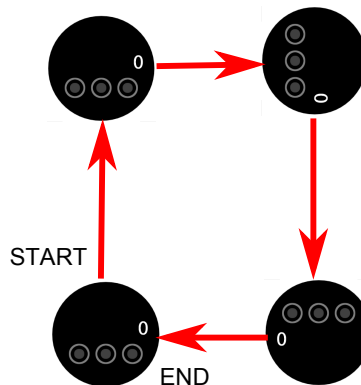
You now have all the information you need to draw a good square!

2 Pseudocode

It's important to think about the problem before we start blindly writing code and hoping for the best. On a piece of paper (a real one; notepad on the computer does NOT count) write down the main steps of the function `square(time)` in plain english.

3 Square!

Now that we're all prepared, we can write our code! Use the skeleton file provided (`SquareDance.py`) as a starting point. Free free to copy the functions you wrote this morning directly into the skeleton file as needed.



4 Scoring

- +10 Robot Moves Forward at least once
- +10 Robot Turns at least once
- +10 Robot Pauses at least once
- +20 Robot moves in a square motion
- 1 Every cm your robot ends away from its start point

Ties will be broken at the judges' discretion.