**Problem Context**

It's the finals of the T20 Cup! Bengaluru and Chennai, neighbours and fierce rivals, are fighting it out for the title. Bengaluru's star batsman Kirat is at the crease. Can he win it for Bengaluru? Build a mobile app to simulate the last 4 overs of the match.

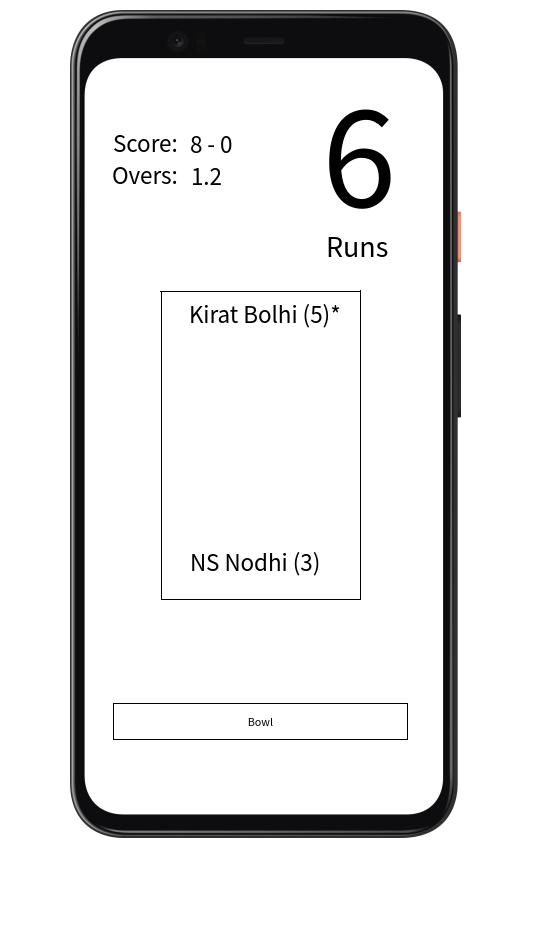
It's the last 4 overs of the match. Bengaluru needs 40 runs to win and with 3 wickets left. Each player has a different probability for scoring runs. Your coding problem is to simulate the match, ball by ball. The match simulation will require you to use a weighted random number generation based on probability to determine the runs scored per ball.



**Rules of the game:**

* Batsmen change strike end of every over. They also change strike when they score a 1,3 or 5
* When a player gets out, the new player comes in at the same position.
* Assume only legal balls are bowled (no wides, no no-balls etc..). Therefore, an over is always 6 balls.

**App wireframe for reference:**



* On pressing the **Bowl** button a new score is generated and added to the current striker. In the above example, Kirat is the striker and his current score is **5**. On pressing the **Bowl** button, if he hits a single, then striker is changed to **NS Nodhi** and the score for **Kirat Bolhi** is updated to **6** and the striker maker (**\***) is switched to **NS Nodhi.**
* A score board with score, wickets and overs is displayed in the top left corner
* The current score of the striker flashes on the top right corner. If the striker gets out then **OUT** is displayed.
* When the match gets over the result is displayed in the screen as follows
  + Bengaluru won the match with 1 wicket and 2 balls
  + Bengaluru lost the match with 2 wickets

**Guidelines**

* Use your favorite language, platform & editor
* We appreciate and look for good quality code.
* Having good coding standards, with proper OO design, responsibility segregation will go a long way in impressing us.
* Great to have -> An extensible and flexible code-base, that won’t require a redesign to address possible future requirements in the given theme.
* Make assumptions where you need to.
* Unit tests are a bonus