

## Why you selected the chosen problem?

I chose this problem because it allowed me to showcase my OOP skills and apply clean code principles effectively. By tackling this specific question, I believed I could demonstrate my ability to solve it with precision and also leave room for future expansion and enhancement of the application.

## Design and approach

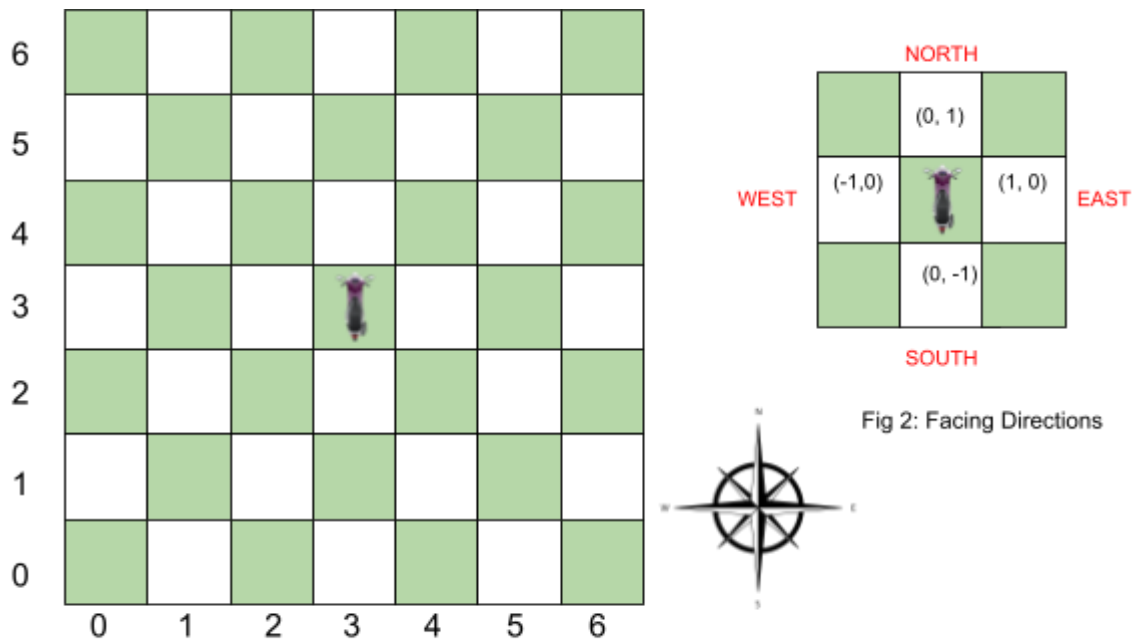


Fig 1: Grid Moving Positions

Fig 2: Facing Directions

### Approach:

I envisioned the movement of the bike on the grid as analogous to navigating a bike on a coordinate plane (x, y) depicted in Fig 1. Additionally, I considered the rotation of each side of the bike as a separate coordinate system, depicted in Fig 2. This approach enabled me to accurately position the bike on the grid and execute precise movements and rotations.

### Design:

I employed a generic Shape interface to represent any plane where a bike or other items could be placed. This approach offers flexibility for future expansion, allowing for the inclusion of additional shapes if desired. To handle the positioning of various objects, including different coordinate systems such as x, y, z, and more complex point structures, I utilized generic Points that can be extended through a Locatable interface.

Moreover, I developed a Generic Simulator and CommandProcessor interface, which can accommodate the implementation of different types of simulators or command processors in the future, if the need arises. To enhance the code's extensibility and adhere to clean code practices, I incorporated the Factory and Strategy patterns. These patterns further facilitate future enhancements and modifications to the codebase.