# Problem B: Not So Self-Driving

Practice Problem

Author: Chris Liu, California High School

## Special Acknowledgement

Unlike all other problems you'll see on our website, this problem was not written by a Lockheed Martin employee. Chris Liu was a student at California High School who attended the Code Quest competition in Sunnyvale, California in 2019. Since then, he has created two of his own programming competitions modeled on the Code Quest competition. Chris is also the lead director for the SRC Hacks hackathons held in 2019 and 2020 in San Ramon, California, which were attended by hundreds of other high school students. The problem below was written by Chris for a programming competition at SRC Hacks, and is presented here with his permission, modified only as needed to fit within our format. Thanks, Chris!

## Problem Background

Dwayne and Johnson are your partners in a group project for your autonomous-vehicles engineering course. However, none of you were paying attention in class, so you forget about the project until the night before it's due! Knowing there would be no time to train a decent self-driving algorithm in just a few hours, you, Dwayne, and Johnson all decide to fake a self-driving car with a few sensors and a bunch of if-statements.

## Problem Description

Your task is to create an obstacle-avoidance system that will output instructions based on information about obstacle distance and the car's speed.

## Sample Input

The first line of your program's input, received from the standard input channel, will contain a positive integer representing the number of test cases. Each test case will include a single line containing two numbers separated by a colon:

- A decimal value V, between 0 and 200 inclusive, denoting the vehicle's current speed (in m/s)
- A decimal value X, between 1 and 400 inclusive, denoting the obstacle's distance from the front of the car (in m)

Both values will be rounded to two decimal places.

```
5
23.15:98.34
2.40:17.33
6.79:5.01
0.00:1.53
113.56:113.56
```

## Sample Output

For each test case, your program must print instructions to the car based on the following criteria:

- If the car is projected to collide with the obstacle in one second or less at the current speed, your program should output SWERVE.
- Otherwise, if the car is projected to collide with the obstacle in five seconds or less at the current speed, your program should output BRAKE.
- Otherwise, your program should output SAFE.

```
BRAKE
SAFE
SWERVE
SAFE
SWERVE
```