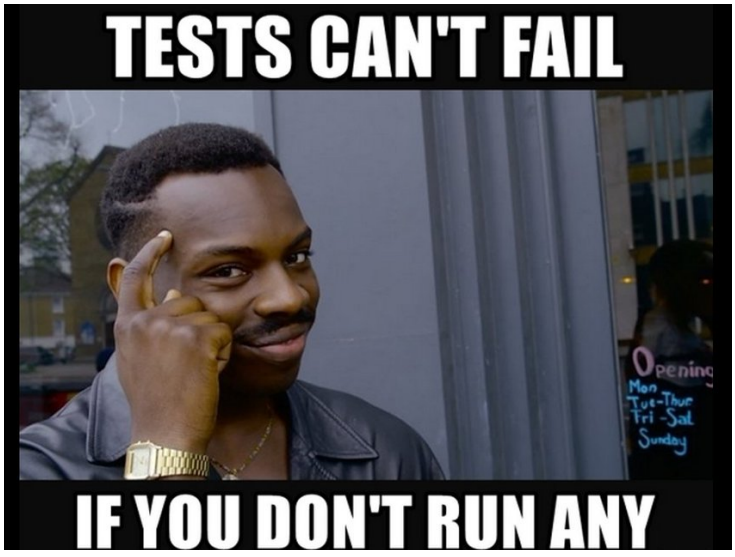
Key Concept: Problem Solving

1. **Understand the Problem.**
    a. Are the pre- and post-conditions clear?
    b. Are there any assumptions we should make?
    c. Do you have enough information about the problem?
    d. Do you fully understand the question?
2. **Solution Design.**
    a. Decompose the problem into separate parts, if possible.
    b. Think about how you would solve the problem manually.
    c. Visualize your thought process.
    d. Walk through a few test cases, especially edge cases.
3. **Code Implementation.**
    a. Translate your thoughts into pseudocode.
    b. Translate pseudocode into actual code.
    c. Optimize your code; follow the DRY principle.
    d. Check your coding style.
4. **Bug Bash.**
    a. Consider all possible branches and edge cases.
    b. Walk through your code with the test cases.



Practice Problem:

Implement a function drawDiamond() that takes a single parameter radius and outputs a string representation of a diamond using stars (*) and dashes (-). See the example test cases below.

```
drawDiamond(1)
-*-
*-*
-*-

drawDiamond(2)
--*--
-*-*-
*---*
-*-*-
--*--

drawDiamond(3)
---*---
--*-*--
-*---*-
*-----*
-*---*-
--*-*--
---*---
```