Home Assignment <11>: Analyzing Test Reports using NumPy with Inheritance

Learning Objective:

The objective of this assignment is to combine NumPy operations with object-oriented programming by implementing inheritance for test execution analysis.

Expected Completion Time:

Best Case: 20 minutes
Average Case: 30 minutes

Assignment Details:

You are working on a QA analytics system. The system stores test execution times (in seconds) using NumPy arrays.

Requirements:

a) Create a base class `TestReport` with an attribute `execution_times` (NumPy array).
b) In `TestReport`, implement methods:

1. `average_time()` → returns the mean execution time.
2. `max_time()` → returns the maximum execution time.
   c) Create a subclass `RegressionReport` that inherits from `TestReport`.

- Add a method `slow_tests(threshold)` that returns all test cases taking more than the threshold time.
  d) In the main section:
- Create a NumPy array with 10 execution times.
- Create a `RegressionReport` object using this array.
- Display average, max, and slow tests using the implemented methods.

Hints to Solve:

- Use `np.mean()` and `np.max()` for calculations.
- Use boolean indexing in NumPy to filter slow tests, e.g., `array[array > threshold]`.
- Use `super().__init__()` to pass the array to the parent class.

Expected Outcome:

Upon completion of this assignment, you should be able to:

- Apply inheritance to reuse base class methods.
- Use NumPy arrays inside class attributes.
- Perform advanced filtering operations with NumPy.
- Connect inheritance with test execution analysis.