

outline the intended structure of the solution (if there are multiple functions/classes involved, etc.).

- iv. Teams should submit their solutions according to supervisor instructions.
- v. If teams are expected to use external libraries, the supervisor will provide the API. Useful external libraries are math, NumPy, and scikit-learn.

4. **SCORING:** High score wins. Written test: 60%, Coding Challenges: 40%.

- a. Coding Challenges will be scored based on correctness (75%) and readability (25%). Intentionally obfuscated code gets 0 points. Any solution that hardcodes the outputs gets 0 points.
- b. Readability entails following major Python conventions, using well-named variables, and organizing code cleanly. The supervisor should be able to easily follow the logic of the code. Teams may add code comments as they see fit.
- c. Tiebreakers (in order of priority): highest Coding Challenges subscore, having well-documented code, first written test question missed.

5. **RECOMMENDED RESOURCES:**

- a. <https://codingbat.com/python>
- b. <https://leetcode.com/problemset/all/>
- c. <https://www.codecademy.com/learn/learn-statistics-with-python>
- d. <https://towardsdatascience.com/beginners-guide-to-machine-learning-with-python-b9ff35bc9c51>
- e. https://www.youtube.com/watch?v=aircAruvnKk&list=PLZHQObOWTQDNU6R1_67000Dx_ZCJB-3pi