

# Customer Requirements from Odyssey Space Research

The purpose of this document is to outline customer requirements provided by Odyssey Space Research to evaluate my Software Engineering capabilities. The requirements have been left deliberately vague in order to make my own decisions about the implementation. It is my objective to impress the senior developers who will evaluate my work.

- A. C++ class that implements a 2 dimensional matrix of form  $n \times m$ , with the following requirements:
  - 1.  $n$  and  $m$  are set by the user when the object is constructed.
  - 2. Overloaded C++ operator:
    - i. Get a specific element of the matrix.
    - ii. Set a specific element of the matrix
    - iii. Access to the matrix is unit-based.
      - The first element of the first row would be referred to as 1 1 not 0 0.
  - 3. The matrix class will support:
    - i. addition
    - ii. scalar multiplication
    - iii. matrix multiplication
      - Given matrices A and B, AB is the dot product of each row of A with each column of B.
      - [https://en.wikipedia.org/wiki/Matrix\\_multiplication](https://en.wikipedia.org/wiki/Matrix_multiplication)
    - iv. At least one other operation of your choice
- B. Unit tests
  - 1. Include them with your submission.
  - 2. Use any test framework you're familiar with, write your own tests.

# Customer Requirements from Odyssey Space Research

- C. The code will be submitted as a zipped git repository.
1. The inherently subjective evaluation will factor many things into the analysis, including:
    - i. Does the code compile?
    - ii. Does it do what it should?
    - iii. Does it pass all unit tests?
    - iv. Are the unit tests sufficient?
    - v. How readable and maintainable is your code?
    - vi. Is your code robust enough to handle any edge cases?
    - vii. Do you exhibit the good habits that would be expected of a professional at your level of expertise?
    - viii. Do you follow (or fail to follow) C++ best practices?
    - ix. What sort of extra features to these bare requirements have you added?
    - x. Is your code efficient?
    - xi. What design and implementation decisions did you make, and why?
      - (You can help by adding relevant comments or other documentation.)