

Cairo Crash Course

Advanced data types - Exercise 4

- An arithmetic series is a sequence of numbers where each term is obtained by adding a fixed number to the previous term. Here for example, each term increases by 2 from the previous term: `[2, 4, 6, 8, 10, ...]`.
- A geometric series is a sequence of numbers where each term is obtained by multiplying the previous term by a fixed number. Here for example, each term is double the previous term `[2, 4, 8, 16, 32, ...]`.

In the following exercises you will have to implement 2 functions to generate `arithmetic_series` and `geometric_series`.

Contract Implementation

Complete the skeletons of the functions `calculate_arithmetic_series` and `calculate_geometric_series` in `advanced_types_exercise_4.cairo` with the following:

1. The function `calculate_arithmetic_series` that takes three parameters:
 - `start`: A u256 that represents the first term of the series.
 - `count`: A u256 that represents the number of terms to generate.
 - `difference`: A u256 that represents the common difference between terms.
2. Initialize an empty array of u256 to store the terms of the arithmetic series.
3. Use a loop to calculate the first `count` terms of the arithmetic series starting from `start` and append each term to the array.
4. Return the array.
5. The function `calculate_geometric_series` that takes three parameters:
 - `start`: A u256 that represents the first term of the series.
 - `count`: A u256 that represents the number of terms to generate.
 - `ratio`: A u256 that represents the common ratio between terms.
6. Initialize an empty array of u256 to store the terms of the geometric series.
7. Use a loop to calculate the first `count` terms of the geometric series starting from `start` and append each term to the array.
8. Return the array.

Writing Tests

1. In the test file `test_advanced_types_4.cairo`, invoke your functions `calculate_arithmetic_series` and `calculate_geometric_series` with different values for

`start`, `count`, `difference`, and `ratio` based on the written TODOs.

2. Implement the necessary assertions to check that each index value in the arrays corresponds to the correct term of the series. If your assertions are correctly implemented, all tests should pass 🎉🥳

Example

- For `calculate_arithmetic_series`, if `start` is 1, `count` is 5, and `difference` is 2, the array should contain: `[1, 3, 5, 7, 9]`.
- For `calculate_geometric_series`, if `start` is 2, `count` is 5, and `ratio` is 3, the array should contain: `[2, 6, 18, 54, 162]`.

Useful links

[Cairo Book: Arrays](#)