Task 27 Spike: ECS Game Loop Management & Scheduling

OPTIONAL

Context

Managing the amount of time and processing power devoted to individual tasks in a game loop is essential when attempting to process large amounts of game data within computational constraints.

Knowledge/Skill Gap:

The developer is unfamiliar with how to manage and schedule operations in a game loop.

Goals

Adapt your Zorkish or SDL spike code so that:

- The time taken by operations performed during the update portion of the game loop can be measured
- An importance metric can be assigned to each operation in the game loop
- The combination of the "time taken" and "importance" values can be used to decide:
  - when in the game loop an operation should take place
  - if it can be skipped for a tick

Expected Output

Repository

1. Code
2. Spike Report

Canvas

1. Spike Report

Notes

- Store operations in a data structure that allows them to be sorted
- Keep track of the total time taken in a loop - if it exceeds a certain value, you should discard all remaining operations for that loop (but increase their importance for the next loop!)
- Your existing spike work may not be complex enough to take demonstrate improvements for this spike - you may need to implement some additional features in order to see the benefit of this spike. Talk to your tutor about implementing this spike in the context of a D/HD project.