**Spike:** 04

**Title:** Maze Game Multithreading

**Author:** Benjamin Fekete

**Goals / deliverables:**

The goal for this lab was to improve upon the grid world game to involve multithreading, allowing for multiple threads to be used, one for rendering and another for processing input.

**Technologies, Tools, and Resources used:**

Tools and resources used

* Visual Studio
* <https://www.geeksforgeeks.org/multithreading-in-cpp/> (for vectors)
* <https://en.cppreference.com/w/cpp/thread/mutex> (unused but for mutex)

**Tasks undertaken:**

* Plan out each thread
* Have an idea on what needs to be optimized and how to implement that
* Research and find examples of threads being used in a similar program

**What we found out:**

I didn’t need to make use of mutex since I didn’t have data being edited in 2 different threads, mutex is only used if there are multiple threads reading AND writing to an object or variable, an example would be a player character being teleported while also having its physics calculated, having 2 threads competing would cause issues.

Threads don’t need to be overly complicated; they can be a simple task if planned out correctly and not just working on it on the fly.

**Recommendations** [Optional – **remove** heading/section if not used!]**:**

* PLAN EVERYTHING, doesn’t need detail but needs a general idea written down on pen and paper to help the thought process, like the order of threads, what causes a thread to be paused/resumed and how threads can interact with each other, the main loop and other data.
* Research more about the topic rather than going in without to much knowledge.