# Low Level Games Programming

## How To Run

1. Open the MemoryManager.sln
2. Inside the main function comment out any unwanted functions
   1. There are functions to demonstrate the memory management, memory pools and ray tracing
   2. If doing raytracing, choose which scene(s) you wish to render by commenting out unwanted ones
3. Run the program to test your selected methods
   1. Note: You can enable/disable verbose output in the “GlobalDefines.h”, by changing the value of “VerboseOutput”
4. If you rendered something using the ray tracer, navigate to “Res/Images/{Name of Scene}” to see the individual .ppm files
   1. You can use ffmpeg to compile into an mp4

## What is Implemented

* Custom Memory Management
  + Global Overloads of new, new[], delete, delete[]
  + Header and Footer for memory allocation
  + Heap Manager and Heaps, including default heap
* Memory Pools
  + Pool manager that manages different size memory pools which pre-allocate blocks of memory which can be used by the overloaded memory management
* Ray Tracer
  + Integrated memory management with the ray tracer application
  + XML Scene Loading
    - Loads an xml scene file using the TinyXML2 library.
    - The scene file allows for the definition of keyframes, which are lerped between to create animations. Any sphere parameter can be lerped
  + There are several precompiled videos of various animations located in the images directory