Students:
DINYAR ISLAM (903586823) - dinyar.islam@gatech.edu
SUSHANT GUHA (903590247) - sguha38@gatech.edu

Element	Percentage Deduction	Details
General functionality Proposed using Multithreading and SFML*: ● Chess piece movement ● Audio playing on piece movement, capture, and game open ● Engine plays move ● Multithreaded engine to particularly larger depth searches for possible moves ● Chess piece capture ● Player has the option to choose the texture of the piece color ● Player has the option to choose the texture of the board ● Player has the option to choose the texture of the pieces ● Letting the player choose their difficulty level ● Initiating appropriate end to game e.g. check mate, stale mate, or draw	Up to 50%	The students delivered on the functionalities of a basic chess game. The functionalities mentioned in the project proposal, or an equivalent replacement functionality are appropriately included.
Custom Class(es) were used	Up to 50%	

Use of one of the five topics: a. Multi-threading std::thread or OpenMP b. OpenGL c. Sockets d. MPI e. GPU (CUDA)	Up to 50%	For example, Multithreading was extensively used for the chess engine. SFML was used.*
Clear Self-Documenting Coding Styles.	10%-25%	This can include incorrect indentation, using unclear variable names, unclear/missing comments, or compiling with warnings.
Extra credit opportunities: - Show the player's next available moves - Pre-moving the game to save time - A game timer that allows for timed games - Move history table - display the past moves played. - Alpha-beta pruning to allow for higher engine depth - Difficulty 4 or above with *reasonable* delay.	Grader's discretion - Up to 5% per functionality added.	The student has included these advanced functionalities on top of the basic requirements of chess.

*Professor Hurley approved use of 2-D graphics using SFML instead of 3-D OpenGL AND the use of the triple happy chess library.