

## Year 8 IST Assignment Three Semester Two (50%) – Game Development

Name: AAA

### Marking Criteria:

Item	Mark	Total
<b>Minimum Requirements (10 marks)</b>		
<ul style="list-style-type: none"><li>Be able to display and track the player's name and numerical score.</li></ul>	2	
<ul style="list-style-type: none"><li>Mostly working set of source code files, which allows the game to be played The game does not crash or provide unusual error messages (for example, in the JavaScript console).</li></ul>	2	
<ul style="list-style-type: none"><li>HTML elements which the user interacts with through the web page, and you manipulate through JavaScript.</li></ul>	2	
<ul style="list-style-type: none"><li>The core game mechanics have been implemented correctly For example in Battleship a random number cannot be less than zero or greater than 5</li></ul>	2	
<ul style="list-style-type: none"><li>Professional presentation of text, images and sounds</li></ul>	2	
<b>Code Structure (20 marks)</b>		
<ul style="list-style-type: none"><li>Appropriate use of functions to minimise repetition of code and to properly organise code.</li></ul>	2	
<ul style="list-style-type: none"><li>Appropriate use of variable scope, using local variables where possible and only using global variables where absolutely necessary.</li></ul>	2	
<ul style="list-style-type: none"><li>Use of appropriate control structures, particular if statements and loops (preferably, for loops where possible).</li></ul>	2	
<ul style="list-style-type: none"><li>Not include repetitive sections of code which could have been avoided through better use of data structures such as arrays or control structures such as loops.</li></ul>	2	
<ul style="list-style-type: none"><li>Appropriately use data structures such as arrays to store data about the game, rather than relying on the user interface state. For example, in a game of noughts and crosses if the game is reliant on parsing the values located within HTML elements to determine the state of the game then your game will not meet this criteria. Instead, the game state should be stored within a programmatic data structure - such as an array.</li></ul>	2	
<ul style="list-style-type: none"><li>Code formatting – indentation, whitespace</li></ul>	2	
<ul style="list-style-type: none"><li>Proper and appropriate naming of all functions, variables and objects</li></ul>	2	
<ul style="list-style-type: none"><li>Code comments – explaining the purpose of the code.<ul style="list-style-type: none"><li>Descriptive and accurate</li><li>Comprehensive and concise</li><li>Demonstrates skill and understanding</li></ul></li></ul>	6	
<b>TOTAL</b>	<b>30</b>	

Item	Mark	Total
<b>Overall Programming Quality (20 marks)</b>		
<ul style="list-style-type: none"> <li>A non-working solution, showing some attempt and minimal or limited understanding of how to code using JavaScript               <ul style="list-style-type: none"> <li>The code is not close to getting the game functioning as designed</li> <li>Understanding of how to use JavaScript is lacking (e.g. code that doesn't compile, variables not defined correctly or in the wrong spot)</li> </ul> </li> </ul>	<b>0-4</b>	
<ul style="list-style-type: none"> <li>A partially working solution, showing substantive attempt and basic understanding of how to code using JavaScript:               <ul style="list-style-type: none"> <li>General game logic has been demonstrated. Overall, the code is somewhat close to getting the game functioning as designed</li> <li>Some essential features work, such as the game loading and the player being able to move</li> <li>Creation of variables for various nodes such as the player, enemies and score display is on the right track</li> <li>Update of variables to implement game mechanics is on the right track</li> <li>A basic level of code comments, with appropriate naming of variables, functions, objects, etc.</li> </ul> </li> </ul>	<b>5-9</b>	
<ul style="list-style-type: none"> <li>A fully working solution showing a thorough understanding of how to code using JavaScript:               <ul style="list-style-type: none"> <li>General game logic has been fully demonstrated. Overall, the code achieves the goal of getting the game functioning as designed</li> <li>All essential features work, such as the game loading and the player being able to move</li> <li>Creation of variables for game components such as the player, enemies and score display is completely functional</li> <li>Update of variables to implement game mechanics and game logic is completely functional</li> <li>Provides thorough and descriptive code comments, with appropriate naming of variables, functions, objects, etc.</li> <li>Appropriate use of control structures to achieve the desired game logic, particularly if statements and loops</li> <li>Appropriate use of functions to minimize repetition of code and to properly organize code</li> </ul> </li> </ul>	<b>10-15</b>	
<ul style="list-style-type: none"> <li>Extension and bonus band (in addition to all of the requirements of a fully working solution) showing an exceptional understanding of how to code using JavaScript:               <ul style="list-style-type: none"> <li>Be the implementation of a complex game which may include graphics manipulation, the implementation of a computer-based AI player, and/or other advanced features.</li> <li>Coding techniques have been masterfully implemented to achieve the complex features (e.g. use of 2D arrays to generate grids, use of objects and functions for a computer based AI player, etc)</li> </ul> </li> </ul>	<b>16-20</b>	
<ul style="list-style-type: none"> <li>Marks awarded</li> </ul>	<b>20</b>	
<b>TOTAL</b>	<b>50</b>	

#### Comments

- AAA