

Ensure that you fill in your assessment, before turning this document in.

Excellent (16-20)	Acceptable (11-15)	Amateur (6-10)	Unsatisfactory (0-5)	TOTAL
<u>DESCRIPTION/LAYOUT/USER INTERFACE</u>				
Excellent visual impact. Layout of objects consistent. Program description turned in on time, thorough and complete.	Interesting visual impact. Layout of objects inconsistent. All objects included in world. Program description turned in on time.	Poor visual impact. Layout of objects unorganized. Missing objects/methods. Program description turned in when program is turned in.	Program description not turned in or is incomplete. Objects in world haphazardly placed. More than 4 objects/methods missing.	17
<u>PROGRAMMING</u>				
Specifications				
Program works and meets all the specifications.	The program works; produces and displays the correct results. The program meets most specifications.	The program produces correct results but does not display them correctly.	The program is producing incorrect results.	18
Requirements				
Program contains all the required elements as listed in the project proposal document.	The program includes most (75-99%) of the required elements, but not all of them.	The program includes about half of the required elements (50-75%), but not all of them.	The program contains few (less than 50%) of the required elements.	18
Readability				
The code is exceptionally well organized and very easy to follow. Variables/objects/methods have meaningful names.	The code is fairly easy to read. Naming of objects consistent.	The code is readable only by an experienced programmer. Little thought to naming of objects.	The code is poorly organized and very difficult to read. Naming of objects is confusing.	15
Comments/Documentation				
Comments are well written and clearly explain what the code is accomplishing and how, without being overly verbose. Every method has header and embedded comments.	Comments are well-written, and every method has header comments that explain the code. Some methods have embedded comments.	Every method has some comments, either header or embedded. Comments may not be useful.	Commenting is sporadic and does not help the reader understand the code. Not all methods have comments.	18
Delivery				
The program was delivered on time.	The program was late.		The program was not turned in.	20
Efficiency				
The code appropriately uses methods to enhance readability of code, array data structures to organize data and loops to simplify code. Event listeners are used appropriately and effectively.	The code organizes some objects into data arrays, but not all that could be. Some loops are used, but some code is repeated. Event listeners are created, but the code written to handle the event does not work properly.	The code has little organization, objects are not organized into arrays, repetition is copied blocks of code in lieu of using loops, event listeners are created but there is no code written to handle the event.	There are no methods, arrays, event listeners, or loops.	19

