

Design	Advantages	Disadvantages
Design 1	<ul style="list-style-type: none"> - Code needs conversion which increases complexity. - typeCoord variable allows conversion and storage 	<ul style="list-style-type: none"> - Instantiating involves the type of coordinates, makes it mildly more complex. - Memory must be allocated to the carriage and coordinates. - Efficiency varies between the typeCoord variable, and the operation called.
Design 2	<ul style="list-style-type: none"> - Code is mildly more complicated as operations to coordinates are involved. - Instantiating is fast since only one set of coordinates. - Memory used is low due to only polar coordinates are stored. 	<ul style="list-style-type: none"> - Efficiency of computations while fast will slower when calculating cartesian coordinates.
Design 3	<ul style="list-style-type: none"> - Code is mildly more complicated as operations to coordinates are involved. - Instantiating is fast since only one set of coordinates. - Memory used is low due to only cartesian coordinates are stored. 	<ul style="list-style-type: none"> - Efficiency of computations while fast will slower when calculating polar coordinates.
Design 5	<ul style="list-style-type: none"> - Code involves abstract classes and subclasses. - Instance would be fast. - Memory usage depends on the subclass used, either low. 	<ul style="list-style-type: none"> - Efficiency depends on the subclass used