E26 Create a table describing the various advantages (pros) and disadvantages (cons) of each of the five design alternatives. Some of the factors to consider are: simplicity of code, efficiency when creating instances, efficiency when doing computations that require both coordinate systems, and amount of memory used.

	PointCP2	PointCP3
Pros	This design only stores 2 Polar points which is more memory efficient compared to the first design The code is simple as it stores and returns 2 Polar points, and will use the given points to calculate and return 2 Cartesian points. There are only 5 instance methods and 2 of the methods just return what the user inputs. The other two methods return a converted point and the final method is just a string method.	This design only stores 2 Cartesian points which is more memory efficient compared to the first design The code is simple as it stores and returns 2 Cartesian points, and will use the given points to calculate and return 2 Polar points. There are only 5 instance methods and 2 of the methods just return what the user inputs. The other two methods return a converted point and the final method is just a string method.
Cons	This is not very efficient when both coordinates are required as Cartesian coordinates are only computed on demand	This is not very efficient when both coordinates are required as Polar coordinates are only computed on demand

E28 Run a performance analysis in which you compare the performance of Design 5, as you implemented it in the previous exercise, with Design 1. Determine the magnitude of the differences in efficiency, and verify the hypotheses you developed in E26.

Both PointCP2 and PointCP3 are similar in efficiency as they both have to convert only two points. When computing larger numbers both designs started to slow down.