	Adv.	Disadv.
Design 1	- Can work with both cartesian and polar coordinates.	- More user input. - More code. - Lots of useless code if you are working with only one coordinate type.
Design 2	- Less code and more readable compared to design 1.	- Can't input/store Cartesian coordinates.
Design 3	- Less code and more readable compared to design 1.	- Can't input/store Polar coordinates.
Design 5	- More organized then any of the above designs.	- Can't input/store both Polar and Cartesian coordinates at the same time.

Which design is best?

- We must compare Polar to Cartesian Conversion time and Cartesian to Polar conversion time for each design.
- We will compare each case using 700000 Polar and Cartesian points.
- We will run and time 3 tests.
- A test consists of converting all points in one test file to it's conterpart and displaying each conversion in the terminal. E.g converting all Polar points in polar_points.csv to there respective Cartesian points and printing out all these cartesian points in the terminal.

Thoughts before experiment:

We believe design 2 and design 3 will be the fastest for Polar to Cartesian and Cartesian to Polar conversion, respectively, since they are both hardcoded to do those conversions whereas design 1 requires additional input and more logic to differentiate between the two conversion. Design5 should have nearly the same, if not, the exact same performance as designs 2 and 3 since there code is written nearly the exact same way.

	Min (ms)	Median (ms)	Max (ms)
Design1/PointCP Converting from Cartesian to Polar (PointCP1CtoPTest.java)	10893	10974	11433
Design3/PointCP3 Converting from Cartesian to Polar (PointCP3CtoPTest.java)	10707	10861	10902
Design5/PointCP3 Converting from Cartesian to Polar (PointCP5-3CtoPTest.java).	10948	11104	11144
Design1/PointCP Converting from Polar to Cartesian (PointCP1PtoCTest.java)	10885	11022	11498
Design2/PointCP2 Converting from Polar to Cartesian (PointCP2PtoCTest.java)	10721	10808	10839
Design5/PointCP2 Converting from Polar to Cartesian (PointCP5-2PtoCTest.java)	10882	10888	10902

Note that all tests where done under similar circumstances, i.e similar background tasks running.

Reflection on results: Based on our results, it would appear that Designs 2 and 3 are the fastest whilst Designs 5 and 1 seems to be equal with a roughly 30-40 ms difference. The difference between the smallest and largest median execution time for Cartesian to Polar and Polar to Cartesian are 243ms and 214ms respectively, a roughly 2% difference. Overall, the difference in performance between all designs is not enough to justify one over another.