

Performance of Analysis:

Run time of test (ms):

Function	Design 1 (Polar)	Design 1 (Cartesian)	Design 2	Design 3	Design 5 (Polar)	Design 5 (Cartesian)
getX()	8800 Min: 8600 Max: 9000	25 Min: 22 Max: 30	240 Min: 239 Max: 240	3 Min: 3 Max: 3	7 Min: 7 Max: 7	8701 Min: 8602 Max: 8942
getY()	8200 Min: 8136 Max: 8235	120 Min: 104 Max: 132	212 Min: 209 Max: 213	2 Min: 2 Max: 2	4 Min: 4 Max: 4	8203 Min: 8153 Max: 8283
getRho()	6 Min: 6 Max: 6	50 Min: 49 Max: 50	2 Min: 2 Max: 2	2 Min: 2 Max: 2	6 Min: 6 Max: 6	6 Min: 6 Max: 6
getTheta()	7 Min: 7 Max: 7	7503 Min: 7484 Max: 7612	2 Min: 2 Max: 2	1202 Min: 1123 Max: 1325	7402 Min: 7324 Max: 7454	7 Min: 7 Max: 7
getDistance()	33401 Min: 33365 Max: 33450	7 Min: 7 Max: 7	805 Min: 802 Max: 808	3 Min: 3 Max: 3	10 Min: 10 Max: 10	33359 Min: 33312 Max: 33413
rotatePoint()	50300 Min: 50285 Max: 50431	33570 Min: 33564 Max: 33582	1310 Min: 1233 Max: 1402	810 Min: 800 Max: 822	33603 Min: 33521 Max: 33764	56234 Min: 53032 Max: 58312
Total Run Time	100400 Min: 100399 Max: 100400	42001 Min: 41230 Max: 42368	3001 Min: 2321 Max: 3101	2001 Min: 1921 Max: 2101	41434 Min: 40812 Max: 43167	106034 Min: 103853 Max: 108211

Description:

Results above were tested using Test, to obtain an accurate data 4 tests were performed, maximum, minimum and median were recorded.

A fixed number of iterations (100,000,00) were used to calculate elapsed time for each method, getX, getY, getRho, getTheta, getDistance, and rotatePoint. It prints out the elapsed time for each method (testPointCP, testPointCP2, testPointCP3, testPointCP5Polar, testPointCP5Cartesian).

Discussion:

In the PointCP class, the "get" method for each coordinate will only perform quickly when the point is created with that type of coordinate. For example, when the point is initialized with polar coordinates, the average time for retrieving rho and theta is 7ms, while retrieving the x and y coordinates takes around 8000ms.

In design 2 and design 3, there is the same issue as in PointCP; however, the time taken to retrieve each coordinate has decreased by around 25ms when computing different types of coordinates.

In design 5, since it is an abstract class with subclasses of design 2 and 3, it has similar results to design 2 and 3.

In conclusion, PointCP is relatively slower than other designs.