E26.

|  |  |  |
| --- | --- | --- |
| Design | Advantage | Disadvantage |
| 1 | * Uses a variable to store the type of coordinate | * When computing a large sample of test cases, Design 1 will use more memory |
| 2 | * Will use less memory | * Is slower because it has to compute twice |
| 3 | * Will use less memory | * Is slower because it has to compute twice |
| 4 | * Will compute quickly because it stores four variables | * Will use more memory because it has to store four variables |
| 5 | * Computes quickly and uses little memory | * Uses too many instances |

E27. Implemented.

E28, E29, E30.

For this test, all designs were tested in one test file called *PointCTestDesign.java*. If the test is run 5 different times:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Design | | | | |
|  | 1 | 2 | 3 | 5 |
| Run 1 (ms) | 29.70737 | 11.25416 | 11.98801 | 10.49884 |
| Run 2 (ms) | 28.72392 | 9.949731 | 14.73122 | 11.66004 |
| Run 3 (ms) | 32.3778 | 9.58397 | 12.65189 | 9.575572 |
| Run 4 (ms) | 32.69831 | 17.26729 | 13.60361 | 9.022266 |
| Run 5 (ms) | 32.2411 | 10.90519 | 7.625934 | 8.112994 |
| Average Computation Speed (ms) | 31.1497 | 11.79207 | 12.12013 | 9.773942 |

Design 1 noticeably takes more time than the other designs. This is consistent with our hypothesis in E26. Design 1 will take an average of 31.15 ms while Design 5 will take an average of 9.774 ms, thus Design 5 is 3.2 times faster than Design 1.

In these tests, there are two parts to consider in each design: Memory usage and computation speed. Both of these are related to each other proportionally so when a program uses more memory, computation speed will also increase.

Design 1, using the most amount of memory out of all the designs in order to store the type variable, will be the slowest design. Design 2 and 3 are very similar as the only operation that takes computation time when it needs to convert from polar to cartesian or vice versa. Thus, Design 2 and 3 will take the same amount of time. Design 5 calls sub-classes depending on the type of variable it is given (it does not store the type in any variable) and it will do no conversions. Thus, it will be the fastest design. In conclusion, the fastest design is one that uses little memory.