Exercise 26

Designs	Pros and cons
Design 1	<u>Simplicity of code</u> This design has a very simple solution which relies solely on two instance variable, the most complex part of this design would actually be the flag. The flag takes time to check the variables to see if they are cartesian or polar. <u>Efficiency when creating variables</u> Immediate designation of variable assignment with a O(1) when going through the variables to determine whether it is polar or cartesian variables <u>Efficiency Doing computations that require both coordinate systems</u> So the pro is that because it has already done the work in checking if it is storing cartesian or polar, right away the system will know whether it will be computed or simply returned <u>Amount of memory used</u> Pro is that the amount of memory is low in size as there are only two instance variables and an extra variable, the flag, will be present. This con is that it uses slightly more space than the designs 2 and 3
Design 2	<u>Simplicity of code</u> Along with Design 3 most simple version of code out of all the designs. <u>Efficiency when creating variables</u> Pro is the Immediate designation of variable assignment which Pro is there is no other steps unlike the other ones so this step will be completed faster than design 1,4, and 5 <u>Efficiency Doing computations that require both coordinate systems</u> So the pro is the efficiency in the case that polar coordinates. (same efficiency as doing computation for design 1 as well) Con is that in the case that cartesian coordinates are computed, the system will need to readjust and compute on demand. This will reduce the overall efficiency. Compared to other designs. <u>Amount of memory used</u> Pro is that this uses the most minimal amount of memory Con is that it wont store the data for cartesian data
Design 3	<u>Simplicity of code</u> Along with Design 2 most simple version of code out of all the designs.

	<u>Efficiency when creating variables</u> Pro is the Immediate designation of variable assignment which Pro is there is no other steps unlike the other ones so this step will be completed faster than design 1,4, and 5 <u>Efficiency Doing computations that require both coordinate systems</u> So the pro is the efficiency in the case that cartesian coordinates. (same efficiency as doing computation for design 1 as well) Con is that in the case that polar coordinates are computed, the system will need to readjust and compute on demand. This will reduce the overall efficiency. Compared to other designs. <u>Amount of memory used</u> Pro is that this uses the most minimal amount of memory Con is that it won't store the data for cartesian data
Design 4 Store both types of coordinates, using four instance variables	<u>Simplicity of code</u> Very simple to code as it only requires two methods with only two parameters(the polar coordinates). In comparison between the first three designs, design 4 will be the least simple as it uses four instance variables unlike the first three designs which uses only 1 pair of instance variables.However, it'll be much more simpler to code compared to design five as no subclasses are implemented. <u>Efficiency when creating variables</u> <u>Efficiency Doing computations that require both coordinate systems</u> <u>Amount of memory used</u> Design 4 will have more memory used when compared to design 3 and 2 as they both only store one type of variable.
Design 5 Abstract superclass	<u>Simplicity of code</u> This one would be the most complicated compared to the other designs we have. Its more complex in that it requires calls to other classes which share a parent class. While this has more potential to be changed around

with designs 2 and 3 as subclasses	and be modular, it sacrifices simplicity in a singular method that the designs prior can achieve. Efficiency when creating variables While the subclass called would depend on what type of coordinates are computed, the system would assign variables at a balanced rate. Unlike other methods which are reliant to a type or memory size, this design is able to assign at the same rate based on the coordinates computed. Efficiency Doing computations that require both coordinate systems Doing the computation will be once again more efficient than the worst case among the prior designs, however it will not be the most efficient among them either. With the first 3 designs, They all have the ability, in best case scenarios, to simply return the immediate computation of whatever coordinate is given. Yet, Design 2 and 3 also have the misfortunes of having to compute on demand without storing if the data doesnt match what they store. So in Amount of memory used