

For this homework assignment you will produce an inheritance hierarchy design for the third programming project. You must produce a high-level representation of your design, in the form of a class diagram. The class diagram must include all the classes you will use to represent the various plants in the simulated ecosystem, and indicate the inheritance relationships among them. Use the notations given in the course notes to indicate inheritance relationships. Your class diagram should only reflect the hierarchy of plant classes.

You must also include documentation for each class in the hierarchy that indicates the planned data and function members of the class, excluding constructors and destructors. Provide a prototype for each such member function, including return type and an indication of which are intended to be virtual. For each member function you must also provide a short description of its responsibilities. If the description requires more than one sentence (or more than two lines of text), that is a good indication the function is doing too much and should be decomposed into components. Indicate whether members are public, protected, or private. You should label any abstract classes as such.

You may place your documentation for each class in a class form if you like, or use any other well-organized format. A class form template is posted on the course website. Separate class forms are not required, nor are operation forms.

The first thing you need to do is read the specification for that project carefully. Pay particular attention to the section that discusses the attributes and behaviors of the various types of creatures that must be simulated.

There are probably many reasonably good designs for this project, and certainly a number of poor ones. A design is not good simply because it is possible to produce a working implementation from it.

You may (should, in fact) decide that you want to make use of additional classes, not identified in the project specification, in order to better organize the inheritance hierarchy. Those classes are part of the hierarchy and should, of course, be represented.

What to turn in and how:

Your design must be submitted in a format that can be opened and edited (to insert comments from the evaluator) in MS Word. An MS Word document is acceptable, as is a plain text file, neatly formatted. The simplest way to draw the class diagram is to use the drawing tools in MS Word. You may use another approach if you like, but bit-mapped drawings (gif, jpeg, bmp, etc.) are not acceptable.

Submit the file containing your design to the Curator System; the assignment will be listed as P3Design. The file should be in a format that can be read in MS Word. Do not zip the file. Instructions for submitting to the Curator are given in the *Student Guide* at the Curator website:

<http://ei.cs.vt.edu/~eags/Curator.html>.

Be sure to follow those instructions carefully. You will submit your design via the URL:

<http://spasm.cs.vt.edu:8080/curator/>

You will be allowed to submit your solution up to three times. Your last submission will be scored. Note: this assignment will be graded by the GTAs, not by the Curator system. You will not be receiving email from the Curator containing your score, although you will probably receive a confirmation message. You should confirm your submission, including its size, on your Curator Home page.

Class description form template:

Class Name	
Purpose	brief description of the role(s) the class is designed to play
States	names and descriptions of logical states objects of this type possess
Attributes	names and descriptions of data members of objects of this type
Operations	prototypes and descriptions of the behaviors supported by objects of this type
Constructors	prototypes and descriptions of the available object construction options
Destructor	description of what the destructor does