Test 2 – during your lab period this week

* same structure as Test 1
* testable material:
	+ Aggregation/Composition
	+ Inheritance
	+ GUI
		- Reading on M-V-C is testable
	+ NO recursion
* To ensure that everyone has space in the lab, GO TO YOUR ASSIGNED LAB SECTION.
* Better instructions will be provided around submitting your work, HOWEVER, you are responsible to know how to submit your work via the command line tools

Recursion

* A recursive method is a method that calls itself
* A recursive method is best used when you can identify a subproblem that is similar to the main problem
* All recursive methods have two things:
	+ Base case: “trivial”, non-recursive solution
	+ Recursive case: calls itself

e.g. Factorial

* This can be broken up as follows:
	+ Factorial(n) = n\*factorial(n-1)
	+ Base case: n = 1, return 1

You can show that

* Recursive methods are correct:
	+ Base case is correct
	+ Assuming that the recursive call returns the right answer, the recursive case is correct
	+ By mathematical induction, the recursion returns the right answer
* Recursive methods halt
	+ (i.e. they eventually hit a stopping case)
	+ Define a “size” that is a nonnegative integer
	+ Show that it decreases with each recursive call
	+ The idea is that the stopping case is the minimum allowed size

Recursive methods on lists

* Find an element in the list (simple search)
* Find the minimum element of a list