Lambda Expressions

Lambda expression: an annonymous fuction.

Callback: A subroutine, passed into a library, that allows the library to "call back" into the main program when appropriate.

The term "lambda expression" comes from the **lambda calculus**, a formal notation for functional programming.

In fuctional programming languages, lambda expressions make it easy to manipulate functions as values—to combine them in various ways to create new fuctions on the fly. This sort of manipulation is less common in imperative languages, lambda expressions can help encourage code reuse ad generality. Examples of callbacks include comparison operator passed into a sorting routine, a predicate used to filter elements of a collection, or a handler to be called in response to some future event.