Homework #2 - Linked Lists

Note: Follow my naming exactly!

Level 1:
Write a singly linked list in Swift (or C++) with the following:

LinkedList
var firstElement : Int?
var lastElement : Int?
var length : Int
func insertAtHead(element: Int)
func insertAtTail(element: Int)
func removeFromHead()
func removeFromTail()
func toArray() -> [Int]

Level 2:
Include the above plus any/all of the following (sliding scale):

LinkedList
func insertAt(index : Int, element : Int)
func removeFrom(index : Int)
func append(list : LinkedList)
func search(element : Int) -> Int //finds first instance of element in list
func searchForAll(element : Int) -> [Int] //returns all indicies of the list that hold this element

Bonus for anyone:
Use generics or templates to allow the LinkedList to hold any data type.

All levels: Please answer the following in a .txt file.
1) When would a linked list be a better choice than a dynamic array?
2) When would a dynamic array be a better choice than a linked list?

C++ properties and method signatures

LinkedList
int getFirstElement()
int getLastElement()
int getLength()
void insertAtHead(int)
void insertAtTail(int)
void removeFromHead()
void removeFromTail()
int* toArray()
void insertAtIndex(int, int) //first one is index
void removeFromIndex(int, int) //first one is index
void appendList(LinkedList*)
int search(int)