

Created by Braeden Moffat	ExampleClass	IntNode	IntNodeArray	Node <Type>	
	Data Members	-nodeData : int -nodePointer : IntNode*	- head : IntNode* - size : int	-nodeData : Type -nodePointer : Node<Type>	
	Methods/Constructors	+getNodeData() : int +getNodePointer() : IntNode* +setNodeData(int) : void +setNodePointer(IntNode *) : void  +IntNode() : constructor +IntNode(int) : constructor +IntNode(int, IntNode*) : constructor	+IntNodeArray(int) : constructor +getSize() : int +getFromIndex(int) : int +setAtIndex(int index, int value) : void	Constructors: +Node<Type>() : constructor +Node<Type>(Type value) : constructor +Node<Type>(Type value, Node<Type>* next) : constructor  Accessors: + getNodeData() : Type + getNodePointer() : Node<Type>* + setNodeData(Type value) : void + setNodePointer(Node<Type> * next) : void	
Timer					
-excecutionTime : clock_t					
+startTimer : void +stopTimer : void +resetTimer : void +displayTimerInformation : void +getExcecutionTimeInMiliseconds : long +Timer() : constructor					
Array<Type>		Model :: List<Type>		BiDirectionalNode<Type>	
-front : Node<Type> * -size : int		-size : int -front : Node<Type> *		-nodeData : Type -nextPointer : BiDirectionalNode<Type> * -previousPointer : BiDirectionalNode<Type> *	
+Array() : default constructor +Array<Type>(int size) : constructor  +~Array<Type>() : destructor +Array<Type>(const Array<Type> & toBeCopied) : copy constructor  +getSize() : int +getFront() const : Node<Type>*		+List<Type>() : constructor +List<Type>(const List<Type> & source) : copy constructor + ~List<Type>() : destructor  +addAtIndex(int index, Type value) : void +add(Type value) : void +remove(int index) : Type +setAtIndex(int index, Type data) : Type +contains(Type data) : bool +getSize() const : int +getFront() const : int +getFront() const : Node<Type> * +getEnd() const : Node<Type> *		+BiDirectionalNode<Type>() : constructor +BiDirectionalNode<Type>(Type value) : constructor +BiDirectionalNode<Type>(Type value, Node<Type>* next, previous) : constructor  +getNodeData() : Type +getNextNodePointer() : BiDirectionalNode<Type>* +getPreviousNodePointer() : BiDirectionalNode<Type>* +setNodeData(Type value) : void +setNextNodePointer(BiDirectionalNode<Type> * next) : void +setPreviousNodePointer(BiDirectionalNode<Type> * next) : void	
+getFromIndex(int index) : Type +setAtIndex(int index, Type value) : void					