Queue		
Queue = {Top= < QueueNode<T>>, Last= <QueueNode<T>>,Size=<Integer>}		
{Inv: size≥0 ∧ size ∈ Z}		
Queue:		→ QUEUE
Add:	QUEUE x T	→ BOOLEAN
Poll:	QUEUE	→ T
Peek:	QUEUE	→ T
IsEmpty:	QUEUE	→ BOOLEAN
GetSize:	QUEUE	→ INTEGER

Constructor Operations:

Queue ()

"Creates a new Queue."

{pre: TRUE}

{post: Stack= {Top = NULL, Last = NULL, Size = 0}}

Modifying Operations:

Add ()

"Adds a new element to the Queue structure linking it to the last position and reassigning the variable last."

{pre: Queue = {Last=<QueueNode<T>>, Size = n}, T=<content>}

{post: Queue = {Last =< QueueNode <T>>, Size=(n+1)}}

Poll ()

"Return the first element that entered to the structure and deletes it from the structure."

{pre: TRUE}

{post: T=<content> if Size ≠0, else NULL}

Analyzing Operation:

Peek ()

"Returns the value of the element first element that entered to the Queue"

{pre: TRUE}

{post: T=<content> if Size ≠0, else NULL}

IsEmpty ()

"Indicates whether the Queue has elements or not."

{pre: TRUE}

{post: TRUE if Size≠0, else FALSE}

GetSize ()

"Returns the number of elements in the Queue structure."

{pre: TRUE}

{post: Size = n}

QUEUENODE		
QueueNode = {Next = < QueueNode <T>>, Prev= < QueueNode <T>>, Content=T}		
{Inv: Content ≠NULL}		
QueueNode:	T	→ QUEUENODE
SetNext:	QUEUE x T	→ QUEUENODE
SetPrev:	QUEUE	→ QUEUENODE
SetContent:	QUEUE	→ QUEUENODE
GetNext:	QUEUE	→ QUEUENODE
GetPrev:	QUEUE	→ QUEUENODE
GetContent:	QUEUE	→ T

Constructor Operations:

QueueNode ()

"Creates a new QueueNode"

{pre: T=<content>}

{post: QueueNode = {Content = T, Next = NULL, Prev = NULL}}

Modifying Operations:

SetNext ()

"Changes the Next QueueNode"

{pre: New QueueNode = {Next = < QueueNode <T>>, Prev= < QueueNode <T>>, Content=T}}

{post: StackNode = {Next = NewStackNode, Prev= <StackNode<T>>, Content=T}}

SetPrev ()

"Changes the Bottom of the StackNode"

{pre: New QueueNode = {Next= < QueueNode <T>>, Prev= < QueueNode <T>>, Content=T}}

{post: QueueNode = {Next= < QueueNode <T>>, Prev= New QueueNode, Content=T}}

SetContent()

"Changes the Content of the StackNode"

{pre: New T=<content>}

{post: QueueNode = {Next= < QueueNode <T>>, Prev= < QueueNode <T>>, Content = NewT}}

Analyzing Operation:

GetNext()

"Returns the Next of the QueueNode"

{pre: TRUE}

{post: QueueNode = {Next= < this.QueueNode <T>>, Prev= < QueueNode <T>>,Content=T}}

GetPrev ()

"Returns the Prev of the QueueNode"

{pre: TRUE}

{post: QueueNode = {Next = this QueueNode, Prev= < this,QueueNode <T>>,

Content = T}}

GetContent ()

"Returns the Content of the QueueNode"

{pre: TRUE}

{post: T=<content>}}