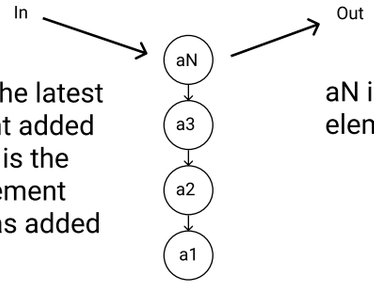
ADT Stack

Stack= { a1,a2,a3... aN}



-aN is the latest element added and a1 is the first element that was added

aN is the top element

- {inv: aN= top}
- Last in is first out

Primitive Operations:

-createStack:	-> Stack
-push: Stack x element	-> Stack
-isEmpty: Stack	-> Boolean
-top: Stack x Element	-> Element
-pop: Stack x Element	-> Element & Stack

Push(T newItem) : Modifier

"Push a new item to the top of the last node vinculed"

{ pre: Stack initialicated }
{ post: Stack size increment +1 and a new item added after the last element: TopElement -> NewElement }

CreateStack( ) : Constructor

"Create a new empty Stack ready to add new items at the top"

{ pre: TRUE }
{ post: NewStack: The new created Stack ready to add new elements }

IsEmpty( ) : Analyzer

"Check if the stack is empty or not and return a boolean value"

{ pre: Stack initialicated }
{ post: True if the Stack is empty or False if isn't empty }

Top( ) : Analyzer

"Returns the element at the beginning of the stack but without removing it from the stack"

{ pre: The Queue isn't Empty and it must stay initialicated }
{ post: Element: The element in the top of the Stack }

Pop( ) : Modifier

"Returns the element to the beginning of the stack and removes it from it"

{ pre: The queue must stay initialicated }
{ post: Element: The Element in the front of the Stack and remove it from it }