

Programación II

TAD Pila

Especificación informal TAD Pila

TAD Pila

VALORES:

- Una pila es una secuencia de cero o más elementos de un mismo tipo $(a_1, a_2, a_3, \dots, a_n)$ donde $n \geq 0$:
 - Si $n=0$ se dice que la pila es vacía.
 - Los elementos de la pila están ordenados de forma lineal, no por su contenido, sino por la posición que ocupan. Es decir:
 - a_i precede a a_{i+1} ($\forall i, i=1 \dots n-1$)
 - a_i sucede a a_{i-1} ($\forall i, i=2 \dots n$)

OPERACIONES (SINTAXIS y SEMÁNTICA)

- Generadoras
 - `createEmptyStack` \rightarrow `Stack`
{*Objetivo*: Crea una pila vacía
Salida: Una pila vacía
Poscondicion: La pila sin datos}
 - `push (Item, Stack)` \rightarrow `Stack`, `Boolean`
{*Objetivo*: Mete un elemento en la pila quedando en la cima
Entrada:
Item: Contenido del elemento a meter
Stack: Pila donde vamos a meter
Salida:
Stack: Pila con el elemento Item en la cima y verdadero si se ha podido meter, falso en caso contrario}

■ Destructoras

- **pop (Stack) \rightarrow Stack**

{*Objetivo*: Saca el elemento de la cima de la pila

Entrada:

Stack: Pila de donde vamos a sacar

Salida:

Stack: Pila sin el elemento de su cima

Precondición: La pila no está vacía}

■ Observadoras

- **peek (Stack) \rightarrow Item**

{*Objetivo*: Recupera el contenido del elemento de la cima de la pila

Entrada:

Stack: Pila donde obtener el dato

Salida:

Item: Contenido del elemento de la cima de la pila

Precondición: La pila no está vacía}

- **IsEmptyStack (Stack) \rightarrow Boolean**

{*Objetivo*: Determina si una pila está vacía

Entrada:

Stack: Pila a comprobar

Salida:

Verdadero si la pila está vacía, falso en caso contrario}