```
#ifndef PILA_H
#define PILA_H

#include "Nodo/Nodo.h"

struct Pila
{
	Nodo* inicio;
	int iTamanio_Pila;
};

//-------------------------------------------------Constructor----------------------------------------------
/*
   PRE: La pila no debe haber sido creada.
   POST: La pila queda creada y el tamanio de la pila queda seteado en 0.
*/
Pila* crearPila();

//-------------------------------------------------Getter---------------------------------------------------
/*
   PRE: La pila debe haber sido creada.
   POST: Devuelve el dato contenido en el iTamanio_Pila de la pila.
*/
int getCantidadDeElementosEnLaPila(Pila*);

//---------------------------------------Funciones de la pila-----------------------------------
/*
   PRE: La pila debe haber sido creada.
   POST: La pila queda vacia y de tamanio 0.
*/
```

```
void vaciarPila(Pila*&);

/*
   PRE: La pila debe haber sido creada.
   POST: Indica si la pila tiene o no elementos que la conpongan.
*/
bool pilaEstaVacia(Pila*);

/*
   PRE: La pila debe haber sido creada.
   POST: Busca un elemento en la pila y me indica en que posicion se encuetra.
*/
int posicionElementoDeLaPila(Pila*, ELEMENTO);

/*
   PRE: La pila debe haber sido creada.
   POST: Agrego al inicio de la pila 1 elemento.
*/
void insertarElementoALaPila(Pila* &, ELEMENTO);

/*
   PRE: La pila debe haber sido creada.
   POST: Almaceno en una variable el ELEMENTO que poseo en el inicio de la pila.
*/
void obtenerElementoDeLaPila(Pila*, ELEMENTO &);

/*
   PRE: La pila debe haber sido creada.
   POST: Remuevo y almaceno en una variable el ELEMENTO que poseo en el inicio
de la pila.
*/
```

```
void eliminarElementoDeLaPila(Pila* &, ELEMENTO &);


#endif // !PILA_H
```