

*Todos deben ser
plantillas*

Objetivo: Familiarizar al estudiante con el concepto y uso de herencia, polimorfismo y excepciones en C/C++.

Enunciado

Resuelva los siguientes problemas. Escriba un código en C/C++ que resuelva los siguientes problemas:

- ✓ 1. Escriba una clase abstracta `LinearDataStructure` que posea como atributo un número entero para la cantidad de elementos contenidos, un método `get` que permita obtener dicho valor y los métodos abstractos `push` y `pop` respectivamente.
2. Herede de la clase `LinearDataStructure` e implemente la clase `Stack` que implementará los métodos abstractos `push` y `pop` respectivamente, como lo haría una pila.
3. Herede de la clase `LinearDataStructure` e implemente la clase `Queue` que implementará los métodos abstractos `push` y `pop` respectivamente, como lo haría una pila.
- ✓ 4. Herede de la clase `exception` e implemente la excepción `EmptyException`.
5. Las clases deben hacer uso de plantillas, para permitir alterar los elementos que se almacenarán en su interior. Además de realizarse una llamado al método `pop` sin que haya ningún elemento que retornar, la clase debe arrojar la excepción `EmptyException`.
6. Escriba un código `main` que solicite al usuario si desea usar una pila (`Stack`) o una cola (`Queue`), cree el objeto polimórfico adecuado y luego pushee los números enteros del 0 al 9. Posteriormente ejecute un ciclo `while` para ir sacando e imprimiendo los elementos del ciclo, capture la excepción para salir del mismo.