

Instrucciones:

Desarrolla y/o documenta una implementación apropiada para las siguientes clases: STACK (lifo), QUEUE (fifo), TABLE/HASH/Dictionary (order).

Las clases deben contener métodos para soportar las principales operaciones de acceso y manipulación (clásicas). NO tienen que ser clases, si el lenguaje de desarrollo no las soporta.

Liga de Github: <https://github.com/danielaramosgarcia/Desarrollo-de-aplicaciones-avanzadas---TTC3002B/tree/main/tarea1>

Descripción de los tests cases:

Stack (Pila).

1. Inserción (Push): Se agregan los elementos 1, 2 y 3 en orden.
2. Impresión del contenido: La salida esperada es: 1 2 3.
3. Extracción (Pop): Debe retornarse el valor 3 y la pila queda como 1 2.
4. Verificación de vaciado: Debe devolver false, ya que aún contiene elementos.
5. Tamaño actual: El valor esperado es 2.
6. Elemento superior: Debe retornar el valor 2.

Queue (Cola)

1. Inserción (Enqueue): Se espera que el orden de la cola sea 10 20 30.
2. Impresión del contenido: La salida debe 10 20 30.
3. Extracción (Dequeue) 2 veces: Los valores retornados deben ser 10 y 20, y la cola debe quedar como 30.
4. Verificación de vaciado: Debe devolver false, ya que aún contiene un elemento.
5. Tamaño actual: El valor esperado es 1.
6. Elemento frontal: Debe devolver el valor 30.

Dictionary (Diccionario)

1. Inserción (Add):
Se agregan tres pares:
 - a. "manzana": "roja"
 - b. "durazno": "naranja"
 - c. "platano": "amarillo"
2. Impresión del contenido: Debe mostrar los tres pares agregados.
3. Eliminación (Remove): Se elimina la clave "durazno".
4. Verificación de vaciado: Debe devolver false, ya que aún tiene dos pares.
5. Tamaño actual: El valor esperado es 2.
6. Búsqueda de clave (ContainsKey): Debe devolver true.
7. Búsqueda de valor (ContainsValue): Debe devolver true.
8. Verificación final de vaciado: Debe seguir devolviendo false.