**CircularDoublyLinkedList**

**Constructor:**

|  |  |  |
| --- | --- | --- |
| Method | Parameter | Description |
| CircularDoublyLinkedList | NONE | Creates an empty circular doubly linked list |

**Functionality:**

|  |  |  |  |
| --- | --- | --- | --- |
| Method | Parameters | Return | Description |
| getContent | NONE | E content | Returns the content of the current node  Exceptions:   * EmptyListException |
| getSize | NONE | int | Returns the size of the list |
| insert | E content | NONE | Inserts a node into the ADT. If the size is > 0, it calls insertAfter |
| insertAfter | E content | NONE | Inserts a node into the ADT after the current node. If the size is 0, it calls insert |
| insertBefore | E content | NONE | Inserts a node into the ADT before the current node. If the size is 0, it calls insert |
| isEmpty | NONE | bool | Returns true if the list is empty, otherwise returns false |
| next | NONE | NONE | Moves current pointer to the next node |
| prev | NONE | NONE | Moves current pointer to the previous node |
| print | NONE | string | Returns a string representation of the linked list. The printout starts with the current element |
| remove | NONE | NONE | Removes the current node and advances the current pointer to the deleted node's next  Exceptions:   * EmptyListException |
| removeAfter | NONE | NONE | Removes the node after the current node  Exceptions:   * EmptyListException |
| removeAll | NONE | NONE | Removes all elements in the ADT  Exceptions:   * EmptyListException |
| removeBefore | NONE | NONE | Removes the node before the current node  Exceptions:   * EmptyListException |
| setContent | E content | NONE | Sets the content of the current node  Exceptions:   * EmptyListException |

**Node:**

**Constructor:**

|  |  |  |
| --- | --- | --- |
| Method | Parameter | Description |
| Node | E content | Creates a node with the content given. The next and previous pointers are set to the created node |
| Node | Node\* prev, E content, Node\* next | Creates a node with the content given. The next and previous pointers are set to the pointers provided. |

**Functionality:**

|  |  |  |  |
| --- | --- | --- | --- |
| Method | Parameters | Return | Description |
| getContent | NONE | E content | Returns the content in the node |
| nextNode | NONE | Node\* next | Returns the pointer to the next node |
| prevNode | NONE | Node\* prev | Returns the pointer to the previous node |
| setContent | E content | NONE | Sets the content of the current node |
| setNext | Node\* next | NONE | Sets the pointer for the next node to the pointer provided |
| setPrev | Node\* prev | NONE | Sets the pointer for the previous node to the pointer provided |