**– public void insertRight(Integer value)**

|  |  |
| --- | --- |
| IF tail is null | O(1) |
| tail is a new Node(value) |  |
| head is equal to tail |  |
| ELSE | O(1) |
| newNode is a new Node(value) |  |
| left node of newNode is equal to tail |  |
| right node of tail is equal to newNode |  |
| tail is equal to newNode |  |
| ENDIF |  |

max(O(1),O(1)= O(1)

**– public Integer removeRight()**

|  |  |
| --- | --- |
| IF tail is null or tail is null | O(1) |
| return null |  |
| ENDIF |  |
| int temp is equal to the value of tail | O(1) |
| IF left node of tail is null | O(1) |
| tail is null |  |
| head is null |  |
| ELSE | O(1) |
| tail is equal to left node of tail |  |
| right node of tail is null |  |
| ENDIF |  |
| return temp |  |

O(1)+ O(1)+max(O(1), O(1)) = O(1)

**– public void insertLeft(Integer value)**

|  |  |
| --- | --- |
| IF head is null | O(1) |
| head is a new Node(value) |  |
| head is equal to tail |  |
| ELSE | O(1) |
| newNode is a new Node(value) |  |
| right node of newNode is equal to head |  |
| left node of head is equal to newNode |  |
| head is equal to newNode |  |
| ENDIF |  |

max(O(1),O(1)= O(1)

**– public Integer removeLeft()**

|  |  |
| --- | --- |
| IF tail is null or tail is null | O(1) |
| return null |  |
| ENDIF |  |
| int temp is equal to the value of head | O(1) |
| IF right node of head is null | O(1) |
| tail is null |  |
| head is null |  |
| ELSE | O(1) |
| head is equal to right node of head |  |
| left node of head is null |  |
| ENDIF |  |
| return temp |  |

O(1)+ O(1)+max(O(1), O(1)) = O(1)

**– public Integer findMinimum()**

|  |  |
| --- | --- |
| Node current is equal to head | O(1) |
| IF current is null | O(1) |
| return null |  |
| ENDIF |  |
| int min is equal to the value of current | O(1) |
| WHILE right node of current is not null | O(N) |
| current is equal to the right node of current |  |
| IF the value of current is less than min | O(1) |
| min is equal to the value of current |  |
| ENDIF |  |
| ENDWHILE |  |
| return min |  |

O(1)+ O(1)+ O(1)+ (O(N)\* O(1)) = O(N)