```python
from exceptions import Empty

class LinkedList:

    class _Node:
        __slots__ = '_element', '_next'

        def __init__(self,element, next):
            self._element = element
            self._next = next

    def __init__(self):
        self._head = None
        self._tail = None
        self._size = 0

    def __len__(self):
        return self._size
    def is_empty(self):
        return self._size == 0

    def add_first(self,e):
        newest = self._Node(e,None)
        if self.is_empty():
            self._head = newest
            self._tail = newest
        else:
            newest ._next = self._head
        self._head = newest
        self._size += 1

    def add_last(self,e):
        newest = self._Node(e,None)
        if self.is_empty():
            self._head = newest
            self._tail = newest
        else:
            self._tail._next = newest
        self._tail = newest
        self._size += 1

    def add_any(self,e,pos):
        newest = self._Node(e,None)
        thead = self._head
        i = 1
        while i < pos:
            thead = thead._next
```

```
48              i += 1
49          newest._next = thead._next
50          thead._next = newest
51          self._size += 1
52
53      def remove_first(self):
54          if self.is_empty():
55              raise Empty('Linked List Empty')
56          value = self._head._element
57          self._head = self._head._next
58          self._size -= 1
59          if self.is_empty():
60              self._tail = None
61          return value
62
63      def remove_last(self):
64          if self.is_empty():
65              raise Empty('Linked List Empty')
66          thead = self._head
67          i = 0
68          while i < len(self) - 2:
69              thead = thead ._next
70              i += 1
71          self._tail = thead
72          thead = thead._next
73          value = thead._element
74          self._tail._next = None
75          self._size -= 1
76          return value
77
78      def remove_any(self, pos):
79          if self.is_empty():
80              raise Empty('Linked List Empty')
81          thead = self._head
82          i = 1
83          while i < pos-1:
84              thead = thead._next
85              i += 1
86          value = thead._next._element
87          thead._next = thead._next._next
88          self._size -= 1
89          return value
90
91      def display(self):
92          thead = self._head
93          while thead:
94              print(thead._element, end='-->')
```

```
 95                 thead = thead._next
 96             print()
 97 
 98 
 99 L = LinkedList()
100 L.add_last(10)
101 L.add_last(20)
102 L.add_last(30)
103 L.add_last(40)
104 L.display()
105 print('Deleted: ', L.remove_first())
106 L.display()
107 L.add_first(70)
108 L.display()
109 print('Deleted: ', L.remove_last())
110 L.display()
111 L.add_any(100,2)
112 L.display()
113 L.remove_any(2)
114 L.display()
115 
116 
117 
118 
119 
120 
121 
122 
123 
124 
```