10. Linked Lists :: Implementation :: DList Class :: remove() Pseudocode

Here is the pseudocode for the *remove*() method:

```
Method remove(In: pIndex) Returns Integer Throws IndexOutOfBoundsException
  -- Obtain a reference to the Node being removed. Note that getNodeAt() throws
  -- an IndexOutOfBoundsException if pIndex is out of bounds.
  node ← getNodeAt(pIndex)

  -- Check to see if we are removing the only element in a list with one element.
  If the size of the list is 1 Then
    Change mHead and mTail to null

  -- Else, check to see if we are removing the head element.
  ElseIf pIndex = 0 Then
    Change the mPrev reference of the Node succeeding node to null
    Change the mHead reference to refer to the Node succeeding node

  -- Else, check to see if we are removing the tail element
  ElseIf pIndex = getSize() - 1 Then
    Change the mNext reference of the Node preceding node to null
    Change the mTail reference to refer to the Node preceding node
```

10. Linked Lists :: Implementation :: DList Class :: remove() Pseudocode

```
  -- Else we removing an element in the interior of the list.
  Else
    Change the mNext reference of the Node preceding node to refer to the Node
    succeeding node
    Change the mPrev reference of the Node succeeding node to refer to the Node
    preceding node
  End If

  Decrement mSize

  Return the data stored at node
End Method remove
```