**Set A Solution**

| **Python** |
| --- |
| def insert\_list(head\_X, head\_A, k):  cur\_X = head\_X.next  cur\_index = 0  prev = cur\_X.prev  while cur\_X != head\_X:  if k == cur\_index:  cur\_A = head\_A  prev = cur\_X.prev  prev.next = cur\_A  while cur\_A != None:  cur\_A.prev = prev  prev = cur\_A  if cur\_A.next == None:  cur\_A.next = cur\_X  cur\_X.prev = cur\_A  break  cur\_A = cur\_A.next  break  cur\_X = cur\_X.next  cur\_index += 1 |
| **Java** |
| void insert\_list(Node head\_X, Node head\_A, int k) {  Node cur\_X = head\_X.next;  int cur\_index = 0;  Node prev = head\_X.prev;  while (cur\_X != head\_X) {  if (k == cur\_index) {  Node cur\_A = head\_A;  prev = cur\_X.prev;  prev.next = cur\_A;  while (cur\_A != null) {  cur\_A.prev = prev;  prev = cur\_A;  if (cur\_A.next == null) {  cur\_A.next = cur\_X;  cur\_X.prev = cur\_A;  break;  }  cur\_A = cur\_A.next;  }  break;  }  cur\_X = cur\_X.next;  cur\_index++;  }  } |

**Set A Rubric**

| **Criteria** | **Marks** |
| --- | --- |
| **Correct condition checking (cur != head)** | 1 |
| **Iterating up to the correct index** | 1.5 |
| **Connecting head of the singly linked list** | 4 |
| **Iterating up to the tail of the singly linked list** | 1.5 |
| **Connecting tail of the singly linked list** | 4 |
| **Making the resulting list doubly** | 3 |
| **Total marks** | **15** |

**Set B Solve**

| **Python** |
| --- |
| def multi\_delete(head):  if not head or not head.next:  return head  cur = head.next  while cur.next != head:  if cur.elem == cur.next.elem:  cur.next = cur.next.next  cur.next.prev = cur  else:  cur = cur.next  return head |
| **Java** |
| void multi\_delete(Node head) {  if (head == null || head.next == head) {  return;  }  Node cur = head.next;  while (cur.next != head) {  if (cur.elem == cur.next.elem) {  cur.next = cur.next.next;  cur.next.prev = cur;  } else {  cur = cur.next;  }  }  } |

**Set B Rubric**

| **Criteria** | **Marks** |
| --- | --- |
| **Correct condition checking (cur != head)** | 1 |
| **Correct condition checking** | 2 |
| **Updating next pointer correctly** | 5 |
| **Updating prev pointer correctly** | 5 |
| **Iterating correctly** | 2 |
| **Total marks** | **15** |