

```

1  #ifndef LLCP_INT_H
2  #define LLCP_INT_H
3
4  #include <iostream>
5
6  struct Node
7  {
8      int data;
9      Node *link;
10 };
11
12 int FindListLength(Node* headPtr);
13 bool IsSortedUp(Node* headPtr);
14 void InsertAsHead(Node*& headPtr, int value);
15 void InsertAsTail(Node*& headPtr, int value);
16 void InsertSortedUp(Node*& headPtr, int value);
17 bool DelFirstTargetNode(Node*& headPtr, int target);
18 bool DelNodeBefore1stMatch(Node*& headPtr, int target);
19 void ShowAll(std::ostream& outs, Node* headPtr);
20 void FindMinMax(Node* headPtr, int& minValue, int& maxValue);
21 double FindAverage(Node* headPtr);
22 void ListClear(Node*& headPtr, int noMsg = 0);
23
24 // prototype of SortedMergeRecur
25 void SortedMergeRecur(Node*& headX, Node*& headY, Node*& headZ);
26
27
28 #endif
29

```