Our Solution(s) Run Code

Solution 1 Solution 2

```cpp
// Copyright © 2020 AlgoExpert, LLC. All rights reserved.

#include <vector>

using namespace std;

class LinkedList {
public:
  int value;
  LinkedList *next;

  LinkedList(int value) {
    this->value = value;
    next = NULL;
  }
};

void recursiveMerge(LinkedList *p1, LinkedList *p2, LinkedList *p1Prev

// O(n + m) time | O(n + m) space - where n is the number of nodes in
// Linked List and m is the number of nodes in the second Linked List
LinkedList *mergeLinkedLists(LinkedList *headOne, LinkedList *headTwo)
  recursiveMerge(headOne, headTwo, NULL);
  return headOne->value < headTwo->value ? headOne : headTwo;
}

void recursiveMerge(LinkedList *p1, LinkedList *p2, LinkedList *p1Prev
  if (p1 == NULL) {
    p1Prev->next = p2;
    return;
  }
  if (p2 == NULL)
    return;
```

Your Solutions Run Code

Solution 1 Solution 2 Solution 3

```cpp
#include <vector>

using namespace std;

// This is an input class. Do not edit.
class LinkedList {
public:
  int value;
  LinkedList *next;

  LinkedList(int value) {
    this->value = value;
    next = NULL;
  }
};

LinkedList *mergeLinkedLists(LinkedList *headOne, LinkedList *headTwo)
  // Write your code here.
  return NULL;
}
```

Our Tests

Custom Output Submit Code

Run or submit code when you're ready.