```
Algorithm  mergeSort(L)
Input: Array of RGB Colors
Output: Sorted Array
   if L.Size() <= 1
      return L

  (left,right) = Partition(L)
   left = mergeSort(left)
   right = mergeSort(right)

 return merge(left, right)

Algorithm merge(left, right)
Input: Array left and right
Output: Merge left and right in sorted order
   result = new array of size n
   leftIndex = 0
   rightIndex = 0

   while leftIndex < left.size() and rightIndex < right.size()
      if (left[leftIndex] = 'red' or left[leftIndex] = 'green') and
         (right[rightIndex] = 'green' or right[rightIndex] = 'blue')

          result.insertLast(left[leftIndex])
         increment leftIndex

      else
         result.insertLast(right[rightIndex])
         increment rightIndex

   insert the remaining elements of left to result
   insert the remaining elements of right to result

   return result
```