给定间隔的集合，合并所有重叠的间隔。

例如，  
给定[1,3],[2,6],[8,10],[15,18]，  
返回[1,6],[8,10],[15,18]。

这道题真的很弱智很弱智！！！！

我的想法是归并排序的思想，每次两两合并，合并结果添加到另一个list中。设置一个boolean的flag，如果当前有合并flag = true，如果没有为false。当没有合并的时候，说明当前为最终结果，退出循环。但是后来想想这种方法是多余的。因为我们可以通过java自带的一些方法来实现更简单的。

我们直接遍历合并就好，合并结果保存在linkedlist中，我们通过getLast()方法每次都能取到最后一个数据单元，之前肯定是合好的数据单元，取到最后一个然后通过比较进行合并。我这里不是很清楚，你使用内在的函数难道就不花时间复杂度吗？？？？以后有机会问问懂的人吧，这里先留个疑问。

下面是代码

public class MergeIntervals {

public static class Interval implements Comparable<Interval>{

int start;

int end;

Interval() {

start = 0;

end = 0;

}

Interval(int s, int e) {

start = s;

end = e;

}

@Override

public int compareTo(Interval o) {

if(this.start>o.start)

return 1;

else if(this.start==o.start)

return 0;

else

return -1;

}

public String toString(){

return "["+start+","+end+"]";

}

}

public static List<Interval> merge(List<Interval> intervals) {

Collections.sort(intervals);//对原来的数据的进行排序，关键字为各个数据单元的start

LinkedList<Interval> temp = new LinkedList<>();//用于记录结果的List

//这里使用LinkedList，这样能哟个geLast()方法直接取到最后一个数据单元

for(Interval item : intervals){

if(temp.size()==0)

temp.add(item);

if (temp.getLast().end>=item.start){//合并，当前一个的end>=后一个的start的时候

//前一个的Interval在temp中

temp.getLast().end = Math.max(temp.getLast().end,item.end);

//合并的时候用比较大的end作为结果

}

else //不用合并直接添加到结果中

temp.add(item);

}

return temp;

}

public static void main(String args[]){

List<Interval> list = new ArrayList<>();

Interval item1 = new Interval(1,9);

Interval item2 = new Interval(2,5);

Interval item3 = new Interval(19,20);

Interval item4 = new Interval(10,11);

Interval item5 = new Interval(12,20);

Interval item6 = new Interval(0,3);

Interval item7 = new Interval(0,1);

Interval item8 = new Interval(0,2);

list.add(item1);

list.add(item2);

list.add(item3);

list.add(item4);

list.add(item5);

list.add(item6);

list.add(item7);

list.add(item8);

List<Interval> res = merge(list);

System.out.print(res);

}

}

体会，直到了这种思路之后，我写上面这段代码的时候，用到了comparable实现自己定义的类的比较，是实现comparaTo方法，同时也看看了它和comarator。还有你要是实现System.out.println直接打印自己定义类的list必须要在自己定义类中实现toString函数，这样当你打印这个类的时候它就会自动调用toString函数。