# 实验一：递归与分治

**一、实验目的**

理解递归算法的思想和递归程序的执行过程，并能熟练编写递归程序。

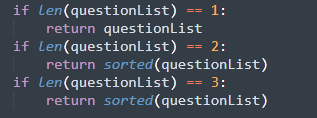
掌握分治算法的思想，对给定的问题能设计出分治算法予以解决。

**二、实验内容**

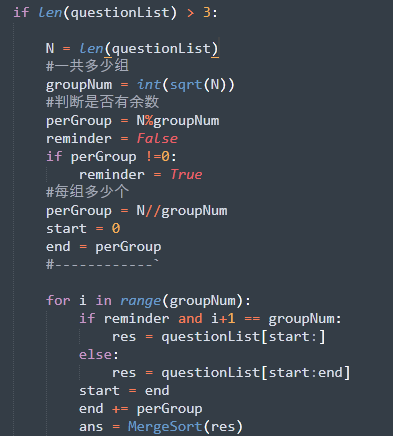
段合并排序算法：

将数组划分为个子数组，每个子数组有个元素。然后递归地对分割后的子数组进行排序，最后将所得到的个排好序的子数组合并排序。

1. **分治算法基本思想**
2. **规模足够小的时候，直接计算返回结果**
3. **将主问题分为根号n个子问题，分别解决每个子问题**
4. **子问题得到解决后，得到的是根号n个有序的数组，将这些数组合并**
5. **实验过程**
6. **小规模问题的处理**



1. **将主问题分成根号n个子问题**



1. **合并**





1. **完整代码**
2. **from** **math** **import** sqrt
3. **import** **random**
4. global\_list = []
5. tmp = []
6. **def** Merge(leftlist,rightlist):
7. i = 0
8. j = 0
9. result = []
10. **while** i < len(leftlist) **and** j < len(rightlist):*#两个列表中拿到小的数*
11. **if** leftlist[i] < rightlist[j]:
12. result.append(leftlist[i])
13. i += 1
14. **else**:
15. result.append(rightlist[j])
16. j += 1
17. *#剩下的数直接放后面*
18. **while** i < len(leftlist):
19. result.append(leftlist[i])
20. i += 1
21. **while** j < len(rightlist):
22. result.append(rightlist[j])
23. j += 1
24. **return** result
25. **def** MergeSort(questionList):
26. **global** global\_list
27. *#规模足够小的时候直接返回*
28. **if** len(questionList) == 1:
29. **return** questionList
30. **if** len(questionList) == 2:
31. **return** sorted(questionList)
32. **if** len(questionList) == 3:
33. **return** sorted(questionList)
34. **if** len(questionList) > 3:
35. N = len(questionList)
36. *#一共多少组*
37. groupNum = int(sqrt(N))
38. *#判断是否有余数*
39. perGroup = N%groupNum
40. reminder = False
41. **if** perGroup !=0:
42. reminder = True
43. *#每组多少个*
44. perGroup = N//groupNum
45. start = 0
46. end = perGroup
47. *#------------`*
48. **for** i **in** range(groupNum):
49. **if** reminder **and** i+1 == groupNum:
50. res = questionList[start:]
51. **else**:
52. res = questionList[start:end]
53. start = end
54. end += perGroup
55. ans = MergeSort(res)
56. **if** ans == None:
57. **continue**
58. global\_list = Merge(ans,global\_list)
59. **实验结果**

