**[求排列的逆序数（分治）](https://www.cnblogs.com/zzyh/p/6625872.html)**

一、题目描述

在Internet上的搜索引擎经常需要对信息进行比较，比如可以通过某个人对一些事物的排名来估计他（或她）对各种不同信息的兴趣，从而实现个性化的服务。

对于不同的排名结果可以用逆序来评价它们之间的差异。考虑1,2,…,n的排列i1，i2，…，in，如果其中存在j,k，满足 j < k 且 ij > ik， 那么就称(ij,ik)是这个排列的一个逆序。

一个排列含有逆序的个数称为这个排列的逆序数。例如排列 263451 含有8个逆序(2,1),(6,3),(6,4),(6,5),(6,1),(3,1),(4,1),(5,1)，因此该排列的逆序数就是8。显然，由1,2,…,n 构成的所有n!个排列中，最小的逆序数是0，对应的排列就是1,2,…,n；最大的逆序数是n(n-1)/2，对应的排列就是n,(n-1),…,2,1。逆序数越大的排列与原始排列的差异度就越大。

现给定1,2,…,n的一个排列，求它的逆序数。

输入

第一行是一个整数n，表示该排列有n个数（n <= 100000)。  
第二行是n个不同的正整数，之间以空格隔开，表示该排列。

输出

输出该排列的逆序数。

样例输入

6

2 6 3 4 5 1

样例输出

8

提示

1. 利用二分归并排序算法（分治）； 与二路归并排序非常相似。
2. 在合并过程中，（low<=i<=mid,mid+1<=j<=high）,当a[i]<=a[j]时，并不产生逆序数；当a[i]>a[j]时，在前半部分比i大的元素都比a[j]大，产生的逆序数个数为mid-i+1.

[复制代码](javascript:void(0);)

1 #include<iostream>

2 #include<cstdio>

3 #include<cstdlib>

4 using namespace std;

5 int n,a[100009],tmp[100009];

6 long long sum;

7 void Merge\_sort(int,int);

8 void Merge\_nx(int,int,int);

9 int main()

10 {

11 scanf("%d",&n);

12 for(int i=1;i<=n;i++)

13 scanf("%d",&a[i]);

14 Merge\_sort(1,n);

15 printf("%d",sum);//输出逆序总数

16 return 0;

17 }

18 void Merge\_sort(int l,int r)

19 {

20 if(l<r)//要有判断

21 {

22 int mid=(l+r)>>1;//\*\*

23 Merge\_sort(1,mid);//归并左边

24 Merge\_sort(mid+1,r);//归并右边

25 Merge\_nx(l,mid,r);//归并排序

26 }

27 }

28 void Merge\_nx(int l,int m,int r)

29 {

30 int z=l,y=m+1;

31 int num=l;// num=l\*\*\*

32 while(z<=m&&y<=r)

33 {

34 if(a[z]<=a[y])

35 {

36 tmp[num++]=a[z++];

37 }

38 else

39 {

40 tmp[num++]=a[y++];

41 sum+=(m-z+1);//逆序总数

42 }

43 }

44 while(z<=m){tmp[num++]=a[z++];}//将剩下的直接放入

45 while(y<=r){tmp[num++]=a[y++];}

46 for(int i=l;i<=r;i++)//l---r

47 a[i]=tmp[i];//将排好的数返回

48 }

[复制代码](javascript:void(0);)