

A.

算法：BFS

最短路径简单题。本题分为3步：（1）通过BFS找到最短路径的大小；（2）在搜索过程中记录每一个节点的前置节点并保存；（3）递归打印最短路径中的坐标点。具体过程见标程！

B.

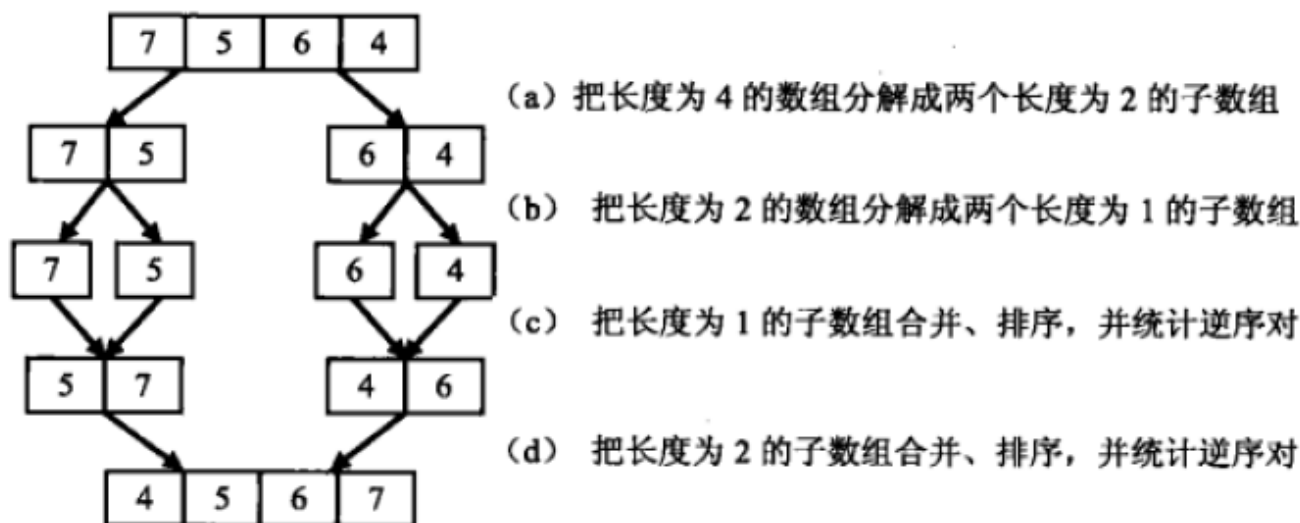
算法：逆序对

仔细读题就会发现是非常简单的计算逆序对的题目（数据范围拒绝“暴力出奇迹”($\odot\odot\odot$)...）！

在数组中的两个数字，如果前面一个数字大于后面的数字，则这两个数字组成一个逆序对。输入一个数组,求出这个数组中的逆序对的总数：

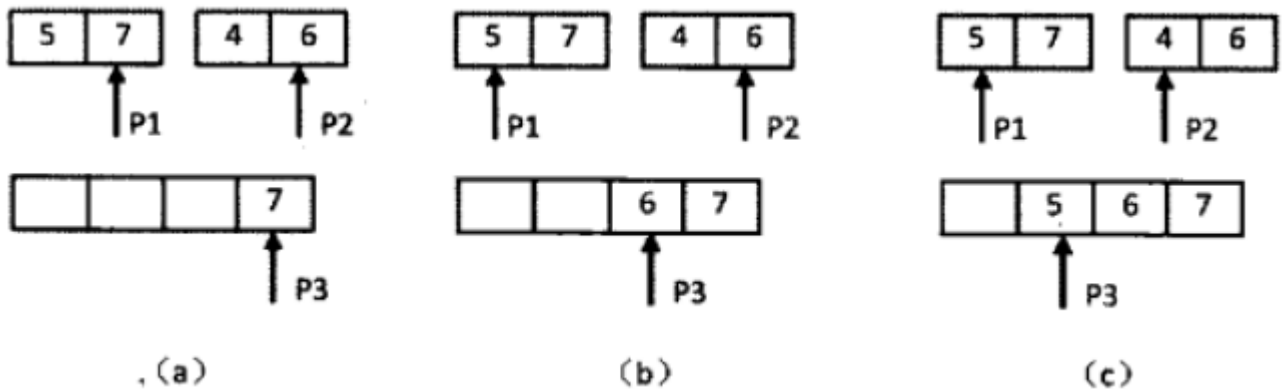
如数组{7,5,6,4}，逆序对总共有5对，{7,5}，{7,6}，{7,4}，{5,4}，{6,4}；

思路2分治思想，采用归并排序的思路来处理，如下图，先分后治：



先把数组分解成两个长度为2的子数组，再把这两个子数组分解成两个长度为1的子数组。接下来一边合并相邻的子数组，一边统计逆序对的数目。在第一对长度为1的子数组{7}、{5}中 $7 > 5$ ，因此(7,5)组成一个逆序对。同样在第二对长度为1的子数组{6}、{4}中也有逆序对(6,4)，由于已经统计了这两对子数组内部的逆序对，因此需要把这两对子数组进行排序，避免在之后的统计过程中重复统计。

逆序对的总数=左边数组中的逆序对的数量+右边数组中逆序对的数量+左右结合成新的顺序数组时中出现的逆序对的数量；



总结统计数组逆序对的过程：先把数组分隔成子数组，先统计出子数组内部的逆序对的数目，然后再统计出两个相邻子数组之间的逆序对的数目。在统计逆序对的过程中，还需要对数组进行排序，其实这个排序过程就是归并排序的思路。

E.

前置知识：二分图最大匹配

做法：以 $i \rightarrow a$ 、 $i \rightarrow b...$ 为边建图（ i 为编号），求最大匹配（匈牙利算法）即可。

考虑匈牙利算法的具体过程：在匹配值为 i 的技能时，那么 1 到 $i - 1$ 的属性肯定已经匹配完成，所以如果 i 对应的编号 j 被匹配了的话，那么就让匹配 j 的那个属性 p 再去找别的物品标号匹配，形象地说，就是用别的物品来释放攻击力为 p 的这个技能，用 j 这个物品释放攻击力为 i 的技能。如果找到这样一条增广路，那么就说明当前可以匹配， $ans++$ 。

F.

考虑容斥

枚举给定素数集的子集，对每个子集求积，将每个积看作因数求出 m 以内所有包含此因数的数的和，积加偶减就搞定了，注意取模

没接触过容斥的可以稍微看一下这篇：<https://blog.csdn.net/tianwei0822/article/details/81781182>

或者看书学习一下