65. 最长上升子序列之高级版

由n个整数组成的数列，记为b[1], b[2], …, b[n]。若存在i1<i2<i3< … < ie 且有b[i1]< b[i2]< … <b[ie]则称为长度为e的上升子序列。求最长上升子序列(Longest increasing subsequence, LIS)

输入格式：

一行，整数序列. 序列长度<=100000, 每个整数绝对值<=10000

输出格式：

最大上升子序列长度

输入样例

2 1 1 2 3

输出样例

3

1924. 最长公共上升子序列

最长上升子序列问题(LIS, Longest Increasing Subsequence)和最长公共子序列问题(LCS, Longest Common Subsequence)都是经典的算法问题。现在我们把两个问题混合，提出最长公共上升子序列问题(LCIS, Longest Common Increasing Subsequence)。

给定2条数字序列A,B, 长度均为n, 求出A和B的公共子序列里保持严格上升最长的子序列长度。

输入文件lcis.in 输入第一行为正整数n, n<=5000。接着两行分别为数字序列A,B,数字之间用空格隔开,每个数字绝对值不超过100000。

输出文件lcis.out 输出一个整数。

输入样例

4

1 3 5 7

5 1 3 7

输出样例

3

**样例说明**

例如1 3 7的长度为3.

输入样例

10

1 5 3 6 3 2 7 3 6 2

9 1 2 3 1 5 3 3 6 1

输出样例

4

**样例说明**

例如1 2 3 6的长度为4.

1925. 炮轰僵尸2

僵尸大战爆发了，你是人类守卫军的将军。眼前有n个僵尸一字排开，守卫军共有m门大炮可以用来炮轰僵尸。其中第i门大炮的位置是在从左到右第s[i]个僵尸的正前方，可以攻击的僵尸必须是连续的一批，个数不能超过l[i]个,而且必须包含第s[i]个僵尸。第i门大炮每轰炸一个僵尸得分p[i]。作为将军，你需要指挥炮手们最优化的分配攻击方案，请问总分最高是多少。当然了，同一个僵尸最多被一门大炮炮轰一次。

输入文件bomb.in 输入第一行为正整数n和m, n<=20000,m<=100。接着m行，每行3个正整数，分别对应l[i],p[i],s[i]。其中l[i],s[i]都在1到n范围内，p[i]均不超过10000。

输出文件bomb.out 输出一个整数。

输入样例

8 4

1 1 7

3 2 2

3 2 3

3 3 5

输出样例

17

**样例说明**

第1门炮位置在7，不安排攻击。

第2门炮位置在2，安排攻击第1,2个僵尸，得分2\*2=4。

第3门炮位置在3，安排攻击第3,4个僵尸，得分2\*2=4。

第4门炮位置在5，安排攻击第5,6,7个僵尸，得分3\*3=9。