一节课入门动态规划

2019.07.27

Hanyuan Shi

这是不可能的！

这个课件讲的是我认知中的动态规划，
它不一定是真的动态规划！

动态规划其实只有两个东西

状态

决策和转移

走向人生巅峰

Problem Statement

Avin 想成为世界无敌的算法大师，他的训练计划一共有 n 天，第 i 天有 a[i] 种训练，选择第 j 种训练可以增长 f[i][j] 的 rating，Avin 每天只能选一种训练。请问 n 天以后 Avin 的 rating 值最高可以到多少?

$1 \le n, a[i] \le 1000$

走向人生巅峰II

Problem Statement

Avin 想成为世界无敌的算法大师，他的训练计划一共有 n 天，一共有 m 种训练可供选择，第 i 种训练需要花费 a[i] 天，结束以后会增长 b[i] 点 rating，一种训练只可以进行一次。请问 n 天以后 Avin 的 rating 值最高可以到多少?

$1 \le n, m \le 1000$

走向人生巅峰III

Problem Statement

Avin 想成为世界无敌的算法大师，他的训练计划一共有 n 天，一共有 m 种训练可供选择，第 i 种训练需要花费 $a[i]$ 天，结束以后会增长 $b[i]$ 点 rating，一种训练可以进行无限次。请问 n 天以后 Avin 的 rating 值最高可以到多少?

$1 \le n, m \le 1000$

Problem Statement

Avin 想成为世界无敌的算法大师，他的训练计划一共有 n 天，一共有 m 种训练可供选择，第 i 种训练需要花费 a[i] 天，结束以后会增长 b[i] 点 rating，这种训练可以进行 c[i] 次。请问 n 天以后 Avin 的 rating 值最高可以到多少?

$1 \le n, m, c[i] \le 1000$

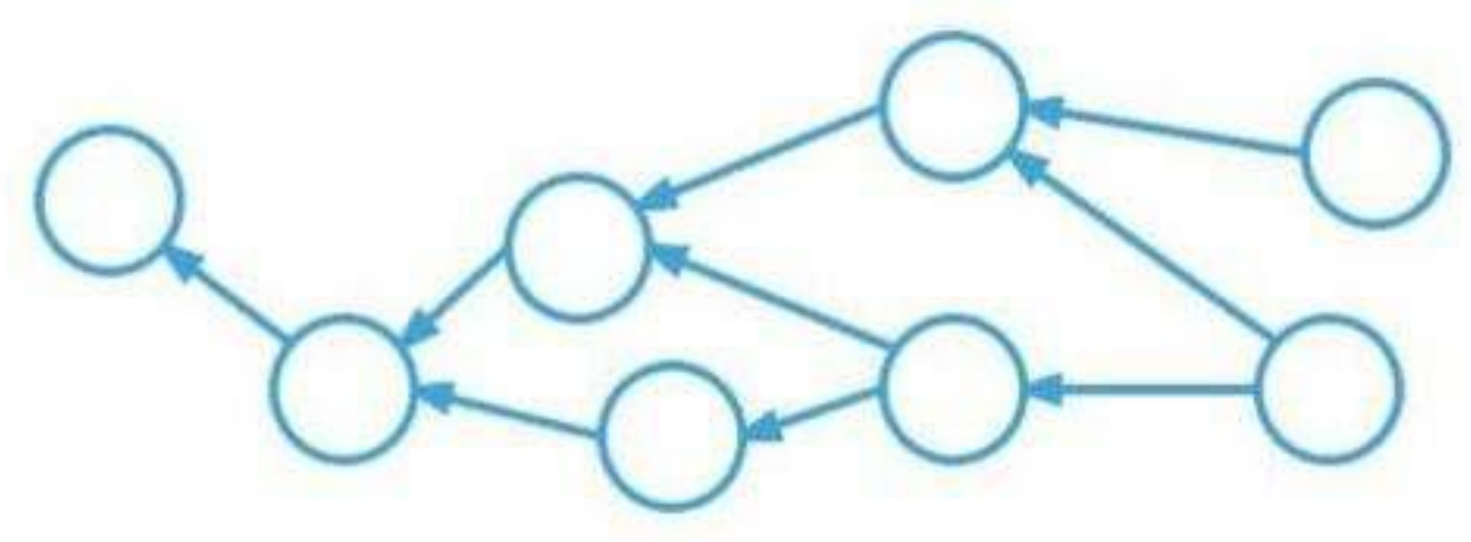
走向人生巅峰V

Problem Statement

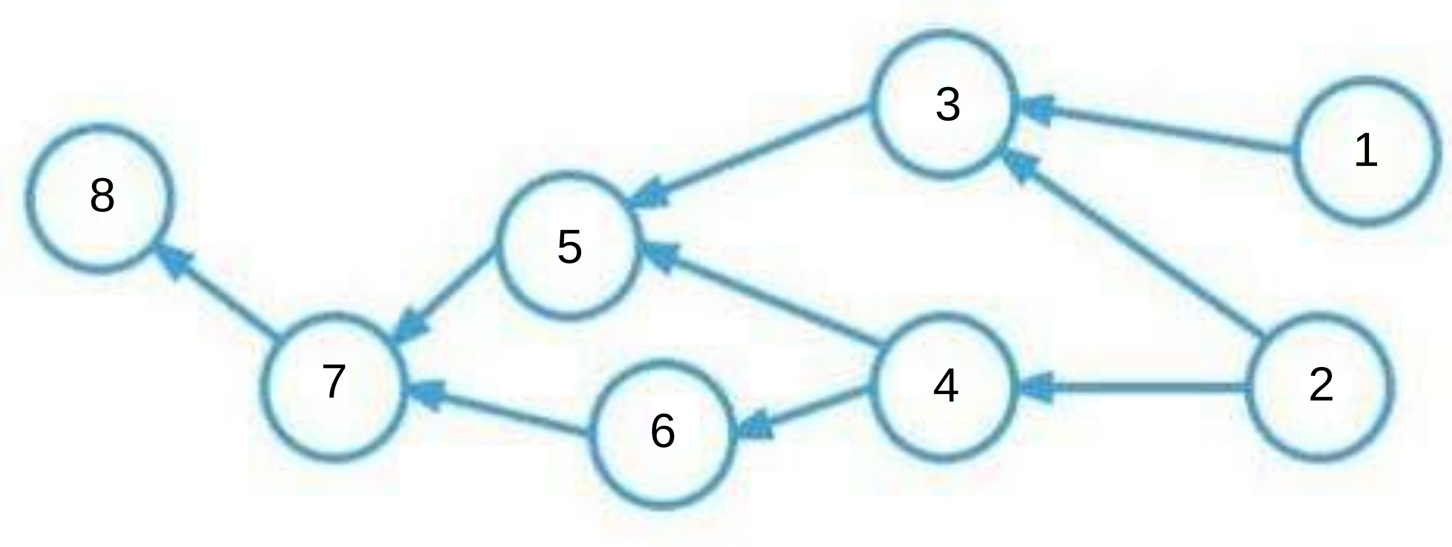
Avin 想成为世界无敌的算法大师，他的训练计划一共有 n 天，一共有 m 种训练可供选择，第 i 种训练需要花费 a[i] 天，结束以后会增长 b[i] 点 rating，但是为了完成第 i 种训练，必须先完成第 c[i] (c[i] < i) 种训练。请问 n 天以后 Avin 的 rating 值最高可以到多少?

$1 \le n \le 1000,\ 1 \le m \le 10$

DAG（有向无环图）



拓扑序



区间dp

Problem Statement

有 n 堆石子排成一排，每堆石子有一定的数量。现要将 n 堆石子并成为一堆。合并的过程只能每次将相邻的两堆石子堆成一堆，每次合并花费的代价为这两堆石子的和，经过 $n-1$ 次合并后成为一堆。求出总的代价最小值。

$1 \le n \le 200$

树形dp

Problem Statement

有一棵 n 个点的树，取第 i 号点会付出 a[i] 的代价。但相邻两个点不能同时不取。问最小代价。

$1 \leq n \leq 100{,}000$

来看点例题吧（开始自闭！）

Problem Statement

You are given a tree consisting exactly of n vertices. Tree is a connected undirected graph with $n - 1$ edges. Each vertex v of this tree has a value a_v assigned to it.

Let $dist(x, y)$ be the distance between the vertices x and y. The distance between the vertices is the number of edges on the simple path between them.

Let's define the cost of the tree as the following value: firstly, let's fix some vertex of the tree. Let it be v. Then the cost of the tree is $\sum_{i=1}^{n} dist(i, v) \cdot a_i$.

Your task is to calculate the **maximum possible cost** of the tree if you can choose v arbitrarily.

Tree

Input

The first line contains one integer n, the number of vertices in the tree ($1 \le n \le 2 \cdot 10^5$).

The second line of the input contains n integers $a_1, a_2, \ldots, a_n$ ($1 \le a_i \le 2 \cdot 10^5$), where a_i is the value of the vertex i.

Each of the next $n - 1$ lines describes an edge of the tree. Edge i is denoted by two integers u_i and v_i, the labels of vertices it connects ($1 \le u_i, v_i \le n$, $u_i \ne v_i$).

It is guaranteed that the given edges form a tree.

Output

Print one integer — the **maximum possible cost** of the tree if you can choose any vertex as v.

Problem Statement

You are given a matrix a, consisting of n rows and m columns. Each cell contains an integer in it.

You can change the order of rows arbitrarily (including leaving the initial order), but you can't change the order of cells in a row. After you pick some order of rows, you traverse the whole matrix the following way: firstly visit all cells of the first column from the top row to the bottom one, then the same for the second column and so on. During the traversal you write down the sequence of the numbers on the cells in the same order you visited them. Let that sequence be $s_1, s_2, \dots, s_{nm}$.

The traversal is k-acceptable if for all i $(1 \le i \le nm - 1)$ $|s_i - s_{i+1}| \ge k$.

Find the maximum integer k such that there exists some order of rows of matrix a that it produces a k-acceptable traversal.

Matrix

Input

The first line contains two integers n and m ($1 \le n \le 16$, $1 \le m \le 10^4$, $2 \le nm$) — the number of rows and the number of columns, respectively.

Each of the next n lines contains m integers ($1 \le a_{i,j} \le 10^9$) — the description of the matrix.

Output

Print a single integer k — the maximum number such that there exists some order of rows of matrix a that it produces an k-acceptable traversal.

String

Problem Description

wls 有一个长度为 n 的字符串，每次他可以将一个长度不大于 l 的子串修改成同一种字母，问至少修改多少次可以使字符串最多含有 k 段。
连续的只含同一种字母的子串被称为一段。比如说， $aaabbccaaa$ 共含有 4 段。

Input

第一行三个整数 n， l， k。
第二行一个字符串。
$1 \leq n \leq 100,000$
$1 \leq l \leq 100,000$
$1 \leq k \leq 10$

中国大学生程序设计竞赛
China Collegiate Programming Contest

WANNAFLY

Problem Statement

You have an array of positive integers $a[1], a[2], ..., a[n]$ and a set of *bad* prime numbers $b_1, b_2, ..., b_m$. The prime numbers that do not occur in the set b are considered *good*. The *beauty* of array a is the sum $\sum_{i=1}^{n} f(a[i])$, where function $f(s)$ is determined as follows:

- $f(1) = 0$;
- Let's assume that p is the minimum prime divisor of s. If p is a good prime, then $f(s) = f(\frac{s}{p}) + 1$, otherwise $f(s) = f(\frac{s}{p}) - 1$.

You are allowed to perform an arbitrary (probably zero) number of operations to improve array a. The *operation of improvement* is the following sequence of actions:

- Choose some number r $(1 \leq r \leq n)$ and calculate the value g = GCD $(a[1], a[2], ..., a[r])$.
- Apply the assignments: $a[1] = \frac{a[1]}{g}$, $a[2] = \frac{a[2]}{g}$, ..., $a[r] = \frac{a[r]}{g}$.

What is the maximum beauty of the array you can get?

GCD

Input

The first line contains two integers n and m $(1 \le n, m \le 5000)$ showing how many numbers are in the array and how many bad prime numbers there are.

The second line contains n space-separated integers $a[1], a[2], \ldots, a[n]$ $(1 \le a[i] \le 10^9)$ — array a. The third line contains m space-separated integers $b_1, b_2, \ldots, b_m$ $(2 \le b_1 < b_2 < \ldots < b_m \le 10^9)$ — the set of bad prime numbers.

Output

Print a single integer — the answer to the problem.

球赛

Problem Description

https://2050.org.cn/games-at2019 是2050非常重要的组成部分之一，包括逐日晨跑、足球风暴、室内骑行挑战、棒球全民打、篮球嘉年华、户外电影等活动。

身体是革命的本钱，这道题是关于运动的。
Alice和Bob在进行乒乓球比赛，比赛一共打了 n 个球，对于每一球，如果Alice赢了，那么裁判员会在计分板上记下'A'，如果Bob赢了则会记下'B'。
时间转眼间到了2050年，计分板上某些信息因为时间流逝丢失了，但我们想要复现当年的激烈局面。
丢失的位置用'?'表示，我们想知道，计分板上对应的乒乓球球赛，最多进行了多少局（最后一局可以没打完，但是如果没打完的话就不计入答案）？
在一局比赛中，先得11分的一方为胜方，10平后，先多得2分的一方为胜方。

Input

第一行一个整数 T $(1 \le T \le 51)$ 表示数据组数。

接下来 T 组数据，每行一个字符串表示计分板上记录的信息，计分板上只包含'A'，'B'，'?'这些字符，计分板长度 $n \le 10000$。

中国大学生程序设计竞赛
China Collegiate Programming Contest

Sum

Problem Statement

Vasya commutes by train every day. There are n train stations in the city, and at the i-th station it's possible to buy only tickets to stations from $i+1$ to a_i inclusive. No tickets are sold at the last station.

Let $\rho_{i,j}$ be the minimum number of tickets one needs to buy in order to get from stations i to station j. As Vasya is fond of different useless statistic he asks you to compute the sum of all values $\rho_{i,j}$ among all pairs $1 \le i < j \le n$.

Input

The first line of the input contains a single integer n $(2 \le n \le 100\,000)$ — the number of stations.

The second line contains n - 1 integer a_i $(i+1 \le a_i \le n)$, the i-th of them means that at the i-th station one may buy tickets to each station from $i+1$ to a_i inclusive.

Output

Print the sum of $\rho_{i,j}$ among all pairs of $1 \le i < j \le n$.

Union

Problem Description

有三个集合 S_1, S_2, S_3，每个集合都是 $\{1, 2, ..., n\}$ 的子集，现有如下条件。
$|S_1| + |S_2| + |S_3| = k$
$|S_1| \ge a_1, |S_2| \ge a_2, |S_3| \ge a_3$
$|S_1 \cup S_2| \ge a_4, |S_2 \cup S_3| \ge a_5, |S_1 \cup S_3| \ge a_6$
$|S_1 \cup S_2 \cup S_3| \ge a_7$
求有多少种集合 (S_1, S_2, S_3) 的有序三元组满足上述要求，请输出答案模 $1,000,000,007$。

Input

第一行输入两个整数 n, k。
第二行输入7个整数 $a_1, a_2 ..., a_7$。
$1 \le n \le 1,000,000$
$1 \le k \le 50$
$0 \le a_i \le 3$

Problem Statement

Sereja has two sequences $a_1, a_2, ..., a_n$ and $b_1, b_2, ..., b_m$, consisting of integers. One day Sereja got bored and he decided two play with them. The rules of the game was very simple. Sereja makes several moves, in one move he can perform one of the following actions:

1. Choose several (at least one) first elements of sequence a (non-empty prefix of a), choose several (at least one) first elements of sequence b (non-empty prefix of b); the element of sequence a with the maximum index among the chosen ones must be equal to the element of sequence b with the maximum index among the chosen ones; remove the chosen elements from the sequences.
2. Remove all elements of both sequences.

The first action is worth e energy units and adds one dollar to Sereja's electronic account. The second action is worth the number of energy units equal to the number of elements Sereja removed from the sequences before performing this action. After Sereja performed the second action, he gets all the money that he earned on his electronic account during the game.

Initially Sereja has s energy units and no money on his account. What maximum number of money can Sereja get? Note, the amount of Seraja's energy mustn't be negative at any time moment.

StringII

Input

The first line contains integers n, m, s, e $(1 \le n, m \le 10^5; 1 \le s \le 3 \cdot 10^5; 10^3 \le e \le 10^4)$. The second line contains n integers $a_1, a_2, \ldots, a_n$ $(1 \le a_i \le 10^5)$. The third line contains m integers $b_1, b_2, \ldots, b_m$ $(1 \le b_i \le 10^5)$.

Output

Print a single integer — maximum number of money in dollars that Sereja can get.

Problem Statement

You have multiset of n strings of the same length, consisting of lowercase English letters. We will say that those strings are easy to remember if for each string there is some position i and some letter c of the English alphabet, such that this string is the only string in the multiset that has letter c in position i.

For example, a multiset of strings {"abc", "aba", "adc", "ada"} are not easy to remember. And multiset {"abc", "ada", "ssa"} is easy to remember because:

- the first string is the only string that has character c in position 3;
- the second string is the only string that has character d in position 2;
- the third string is the only string that has character s in position 2.

You want to change your multiset a little so that it is easy to remember. For a_{ij} coins, you can change character in the j-th position of the i-th string into any other lowercase letter of the English alphabet. Find what is the minimum sum you should pay in order to make the multiset of strings easy to remember.

StringIII

Input

The first line contains two integers n, m $(1 \le n, m \le 20)$ — the number of strings in the multiset and the length of the strings respectively. Next n lines contain the strings of the multiset, consisting only of lowercase English letters, each string's length is m.

Next n lines contain m integers each, the i-th of them contains integers $a_{i1}, a_{i2}, \dots, a_{im}$ $(0 \le a_{ij} \le 10^6)$.

Output

Print a single number — the answer to the problem.

Rento

题目描述

wls等六人在玩大富翁。

大富翁的棋盘上一共有n块地，这些地围成了一个圈（n号地的下一块地是1号地）。

每一轮，六个玩家会依次掷出一颗骰子（编号小的玩家先掷），并往前走骰子上显示的数字那么多步。

第一个踩到某块地的玩家能够得到这块地。

现在请问，500轮以后，每个玩家拥有的地的块数的期望分别是多少？

所有人都从1号点出发，出发前大家不能买地（第二次到1号点才能买这块地）。

Rento

输入描述

第一行一个整数n。

接下来6行，每行6个两位小数，$p[i][j]$表示第i个人掷出j的概率。

$1 \le n \le 500$

对于所有的i，满足$\sum p[i][j] = 1$

输出描述

输出6个小数，第i个数表示i号玩家拥有的地的块数的期望。

答案保留3位小数。

Eat

题目描述

wls在玩一个游戏。

wls有一个n行m列的棋盘，对于第i行第j列的格子，每过$T[i][j]$秒会在上面出现一个糖果，第一次糖果出现在第$T[i][j]$秒，糖果仅会在出现的那一秒存在，下一秒就会消失。

假如wls第k秒在第i行第j列的格子上，满足$T[i][j]|k$，则wls会得到一个糖果。

假如当前wls在第i行第j列的格子上，那么下一秒他可以选择往上下左右走一格或停在原地。

现在请问wls从指定的S出发到达指定的T，并且在路上得到至少C个糖果最少需要多少时间?

wls在S的初始时间为第0秒。

Eat

输入描述

第一行三个整数n，m，C。

接下来n行，每行m个整数$T[i][j]$。

接下来四个整数xs, ys, xt, yt，其中(xs, ys)表示S，(xt, yt)表示t。

$1 \le n, m, T[i][j] \le 10$

$1 \le C \le 10^{18}$

$1 \le xs, xt \le n$

$1 \le ys, yt \le m$

动态dp

Problem Statement

有一棵 n 个点的树，取第 i 号点会付出 a[i] 的代价。但相邻两个点不能同时不取。有 m 个操作，每次操作约定了一个点必须取或必须不取，问每次操作后的最小代价。

$1 \le n, m \le 100{,}000$

Solution

考虑一条链的情况

$$f_{i,0} = f_{i-1,1}$$
$$f_{i,1} = a_i + min(f_{i-1,0}, f_{i-1,1})$$

写成矩阵形式，乘运算定义为加法，加运算定义为取min

$$\begin{bmatrix} \infty & 0 \\ a_i & a_i \end{bmatrix} \begin{bmatrix} f_{i-1,0} \\ f_{i-1,1} \end{bmatrix} = \begin{bmatrix} f_{i,0} \\ f_{i,1} \end{bmatrix}$$

线段树维护之。

Solution

回到一棵树

$$A_u = \sum_{v是u的轻儿子} f_{v,1}$$

$$B_u = a_u + \sum_{v是u的轻儿子} min(f_{v,0}, f_{v,1})$$

写成矩阵形式

$$\begin{bmatrix} \infty & A_u \\ B_u & B_u \end{bmatrix} \begin{bmatrix} f_{mson,0} \\ f_{mson,1} \end{bmatrix} = \begin{bmatrix} f_{u,0} \\ f_{u,1} \end{bmatrix}$$

树链剖分维护之。

谢谢！

GL && HF！