SOI 2019

第一试

|  |  |  |  |
| --- | --- | --- | --- |
| 中文题目名称 | 简单的数据结构 | 组合数问题 | 小乔 |
| 英文题目名称 | A | number | xiaoqiao |
| 输入文件名 | A.in | number.in | xiaoqiao.in |
| 输出文件名 | A.out | number.out | xiaoqiao.out |
| 每个测试点时限 | 4秒 | 1秒 | 1秒 |
| 内存限制 | 512MB | 512MB | 512MB |
| 测试点数目 | 50 | 20 | 20 |
| 每个测试点分值 | 2 | 5 | 5 |
| 结果比较方式 | 全文比较（过滤行末空格及文末回车） | | |
| 题目类型 | 传统 | 传统 | 传统 |

提交源程序须加后缀

|  |  |  |  |
| --- | --- | --- | --- |
| 对于Pascal语言 | A.pas | number.pas | xiaoqiao.pas |
| 对于C 语言 | A.c | number.c | xiaoqiao.c |
| 对于C++ 语言 | A.cpp | number.cpp | xiaoqiao.cpp |

注意：最终测试时，所有编译命令均不打开任何优化开关。

简单的数据结构

【问题描述】

小A在一场模拟赛中发现别人直接粘自己写的板子库，里面Trie树，AC自动机，线段树应有尽有，并且那场模拟赛他尽然还AK了，十分眼红，便想写一个模板库，让他也AK每一场模拟赛。

于是，他准备先写个简单的板子作为AK模拟赛的第一步。

给定一棵个节点的树，初始时该树的根为树的重心(若大于个，便取序号最小的)，每个节点有一个给定的权值。

下面依次进行个操作，操作分为如下6种类型：

换根：将一个指定的节点设置为树的新根。

修改路径权值：给定两个节点，将这两个节点间路径上的所有节点权值（含这两个节点）增加，（或乘上、修改为）一个给定的值。

修改子树权值：给定一个节点，将以该节点为根的子树内的所有节点权值增加（或乘上、修改为）一个给定的值。

询问路径：询问某条路径上节点的权值和。

询问子树：询问某个子树内节点的权值和。

询问两点的：询问两个节点的最近公共祖先。

小A看到了正在勇敢拼搏的你，于是准备一个十分有用的工具（详情见提示）。

【输入格式】

第行两个整数，分别表示节点的个数，操作的个数。

第行个整数表示第个节点的初始权值。

第行到第行，每行两个整数，表示有从到的边。

接下来行,每行第一个整数表示操作的类型

若类型为，则接下来一个整数，表示新根的编号。

若类型为，则接下来三个整数与一个字符串，，分别表示路径两端的节点编号以及增加（或乘上、修改为）的权值。其中若为，为增加，若为，为乘上，若为为，则为修改为，其中类型如此类推。

若类型为，则接下来两个整数与一个字符串，分别表示子树根节点编号以及增加（或乘上、修改为）的权值。

若类型为，则接下来两个整数，表示路径两端的节点编号。

若类型为，则接下来一个整数，表示子树根节点编号。

若类型为，则接下来两个整数，表示两个查询节点。

其中所有查询都对取模。

【输出格式】

对于每一个类型为或或的操作，输出一行一个整数表示答案。

【样例输入】

5 5

5 3 1 4 3

1 2

2 3

3 4

4 5

5 1

4 2 3

4 5 2

3 1 1 mul

1 4

【样例输出】

5

4

11

【提示】

checker.exe文件是小A专门为你准备的AK模拟赛的工具，是标程的exe文件。

请注意常数因子带来的程序效率上的影响与对于取模的要求。

因为测试点较多，时间较大，若不能完成一个子任务，请特判一下退出。

【数据规模与约定】

对于每个子任务，前一半的数据均无操作。

对于的数据，。

对于的数据，。

对于的数据，。其中的数据，数据是一条链，满足。

对于全部数据，,。

组合数问题

【题目背景】

小B因为数学成绩不好，被班上众神仙虐，所以心情十分悲伤，直到有一天有了一个别开生面的数学课。

【问题描述】

在课上，老师分别提出了3个问题。

Question1给定，求的值。

Question2 给定，求的值。

Question3 给定，求的值。

全班同学除了小B都沉默了，而小B却都知道此做法，于是作为班上的众多神仙之一的你，请证明你的数学能力。

【输入格式】

第一行包含两个整数 ()。其中代表数组长度，代表是3个问题中哪个问题。

第二行包含个整数，代表。

第三行包含个整数，代表。

【输出格式】

输出一个整数，表示的答案。答案对取模。

【样例输入1】

1 1

2 3

【样例输出1】

10

【样例输入2】

3 2

1 1 2

1 1 1

【样例输出2】

26

【样例输入3】

3 3

1 1 2

1 1 1

【样例输出3】

26

【数据规模与约定】

数据点1-2满足，，。

数据点3-4满足，，。

数据点5-6满足，，。

数据点7-10满足，，。

数据点11-20满足，，。

对于所有数据满足，。

小乔

【问题描述】

恋之微风·小乔，是手游《王者荣耀》中的法师型英雄，擅长远程消耗。小乔有一把神奇的扇子，借助灵活的走位可以对敌人造成高额的伤害。小乔是小A最喜欢（会玩）的英雄之一。在一场梦里，小A与小乔来到了一个异次元世界。

异次元世界位于极坐标系中。小乔定义了一个值，以等分弧度（详见样例）。小乔利用她神奇的扇子，进行次“绽放之舞”操作。对于第次“绽放之舞”操作，小乔将设定半径，起始位置，终止位置，她借助自己神奇的扇子，以坐标系原点为圆心，为半径，将圆心角到圆心角这部分**扇形**区域**逆时针**叠加一层“治愈微笑”。

小乔想到了一个有趣（奇怪）的问题，她希望知道有多大面积的区域被叠加过至少层“治愈微笑”。这个问题难倒了平日里善于发现并解决问题的小A，现在小A求助于你，希望你能帮他解决这个问题。

我们设答案的值为，为了方便表达，你只需要输出（可以证明这是一个非负整数）的值即可。

【输入格式】

第一行是三个整数，，。

接下来行，依次描述每个“绽放之舞”操作，每行包含三个整数，，。

【输出格式】

输出只包含一个整数，表示的值。

【样例输入】

3 8 2

1 -8 8

3 -7 3

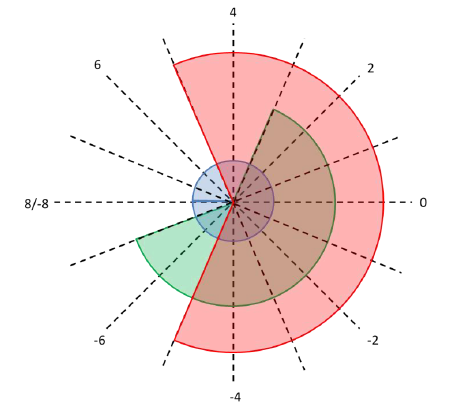
5 -5 5

【样例输出】

76

【样例说明】

样例中的内容如下图所示：



【数据规模与约定】

所有测试点的数据规模与约定如下：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 测试点编号 | 的规模 | 和的规模 | 的规模 | 约定 |
| 1 |  |  |  |  |
| 2 |
| 3 |  |
| 4 |
| 5 |
| 6 |
| 7 |  |  | 所有的都相等 |
| 8 |
| 9 |
| 10 |  |
| 11 |
| 12 |
| 13 |  |  | 无 |
| 14 |
| 15 |  |
| 16 |
| 17 |  |  |
| 18 |
| 19 |  |
| 20 |

对于全部测试数据满足，。