

コンテスト (Contest)

世の中には様々なプログラミングコンテストがある。例えば、情報オリンピック、ACM-ICPC、ICFPの様に世界中の相手と競い合う物や、Supercomputing Contest、パソコン甲子園、EPOCH @まつやまの国内大会などなど、枚挙にいとまがない。

そのようなコンテストに参加していると自分でコンテストを開催したいと考える人もあらわれる。あなたの友人もそんな一人である。友人はコンテストの問題準備に忙しく、コンテストのログから参加者の得点を計算するプログラムの作成をあなたに依頼してきた。是非とも友人のコンテストが成功するように、間違いのない得点計算プログラムを作ってあげて欲しい。

コンテストの参加者は N 人で、問題が M 問で、競技時間が T 秒である。参加者には 1 から N までの番号が割り振られている。問題にも 1 から M までの番号が振られている。

このコンテストでは、各参加者は問題毎に、問題文を閲覧し、問題を解き、解答のソースコードを提出する。競技は時刻 0 からはじまり、時刻 T で終了する。参加者が問題を閲覧すると、時刻と参加者番号と問題番号がログに 1 件の情報として記録される。参加者が解答ソースコードを提出すると、時刻と参加者番号と問題番号と解答ソースコードの正誤がログに 1 件の情報として記録される。

各参加者はコンテスト開始から 1 秒毎に「ある問題を閲覧する」か「ある問題のソースコードを提出する」のいずれかを 1 回行うことができる。閲覧前にその問題のソースコードを提出する事や、正解後に再度ソースコードを提出する事は出来ない。

参加者 j が問題 i を、時刻 S_{ij} に閲覧し、 w_{ij} 回の不正解のソースコードを提出後、時刻 E_{ij} に正解のソースコードを提出したときの得点は、

$$(p_i - (E_{ij} - S_{ij}) - 120w_{ij}) \text{ と } X \text{ のうち大きい方}$$

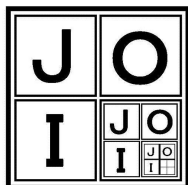
である。ただし、 p_i は問題 i の配点、 X は問題番号に依存しない正解した時の最低得点である。

競技終了までに、正解しなかった問題の得点は 0 点である。

各参加者の得点は、その参加者の問題毎の得点の合計である。

課題 (TASK)

N, M, T, X , ログに含まれる情報の件数、各問題の配点、および、ログの内容が与えられたときに各参加者の得点を計算するプログラムを作成せよ。



制限 (CONSTRAINTS)

$1 \leq N \leq 1,000$	参加者数
$1 \leq M \leq 10$	問題数
$100 \leq T \leq 10,000$	コンテストの競技時間
$1 \leq X \leq 100$	正解問題の最低得点
$0 \leq Y \leq 10,000$	ログに含まれる情報の件数
$X \leq p_i \leq T$	問題 i の配点
$0 \leq t_1 \leq t_2 \leq \dots \leq t_k \leq \dots \leq t_Y \leq T$	ログの k 番目の情報が記録された時刻を表す整数
$1 \leq n_k \leq N$	ログの k 番目の情報に含まれる参加者番号
$1 \leq m_k \leq M$	ログの k 番目の情報に含まれる問題番号

入力 (INPUT)

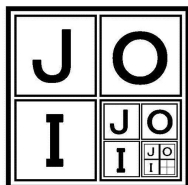
標準入力から以下の入力を読み込め。

- 1 行目には、参加者数 N 、問題数 M 、競技時間 T 、正解問題の最低得点 X 、ログの件数 Y が空白を区切りとして書かれている。
- 続く M 行のうち i 行目には、問題 i の得点を表す整数 p_i が書かれている。
- 続く Y 行はログの内容を表す。それらのうちの k 行目にはログの k 番目の情報が書き込まれており、その情報が記録された時刻を表す整数 t_k 、参加者番号を表す整数 n_k 、問題番号を表す整数 m_k 、および “open”, “correct”, “incorrect” のいずれかの文字列が空白区切りで書かれている。
“open” は問題を閲覧したことを表す。“correct” は提出したソースコードが正解であったことを表す。
“incorrect” は提出したソースコードが不正解であったことを表す。

出力 (OUTPUT)

標準出力に以下のデータを出力せよ。

- データは N 行からなり、 j 行目は j 番の参加者の得点を含む。



入出力例 (EXAMPLE)

入力例 (Sample Input)	出力例 (Sample Output)
1 3 1200 30 8	410
50	
1000	
800	
0 1 1 open	
100 1 2 open	
400 1 1 correct	
500 1 2 incorrect	
600 1 2 correct	
700 1 3 open	
800 1 3 incorrect	
900 1 3 incorrect	

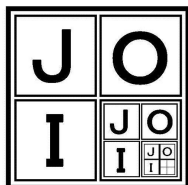
1 番の参加者は、1 問目で 30 点、2 問目で 380 点、3 問目で 0 点を獲得する。2 問目の得点は

$$1000[\text{問題 2 の配点}] - (600[\text{正解時刻}] - 100[\text{閲覧時刻}]) - 120 \times 1[\text{不正解, 1 回}]$$

で計算される。

入力例 (Sample Input)	出力例 (Sample Output)
3 2 1000 10 10	392
878	1390
712	778
0 2 1 open	
0 1 1 open	
0 3 1 open	
100 3 1 correct	
100 2 1 correct	
100 1 2 open	
200 2 2 open	
200 1 2 incorrect	
300 1 2 correct	
300 2 2 correct	

参加者が複数の場合、同じ時刻に複数の情報が記録されている場合がある。

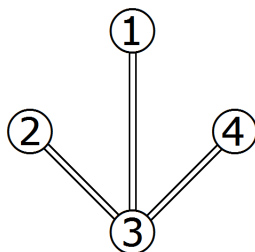


高速道路 (Highway)

カナダには、都市 1 から都市 N までの N 個の都市と、高速道路 1 から高速道路 $N-1$ までの $N-1$ 本の高速道路がある。すべての高速道路は 2 つの都市を結んでおり、どちらの方向にも通ることができる。さらにこの高速道路網は、どの 2 つの都市もいくつかの高速道路を乗り継ぐことで行き来できるように設計されている。言い換えると、 N 個の都市と $N-1$ 本の高速道路は木構造をなしている。

渋滞情報センターの施設長に任命されたあなたは、この高速道路網の渋滞情報を管理しなければならない。

この施設では、 $N-1$ 本それぞれの高速道路について、その始点の都市から終点の都市までの所要時間のデータを管理している。たとえば下図のように都市 1 と都市 3、都市 3 と都市 4、都市 2 と都市 3 が高速道路で繋がっている状況の場合、 $(i, j) = (1, 3), (3, 1), (3, 4), (4, 3), (2, 3), (3, 2)$ のそれぞれについて「都市 i から都市 j への所要時間」をこの施設で管理している (高速道路は上りと下りで所要時間が同じとは限らないことに注意せよ)。



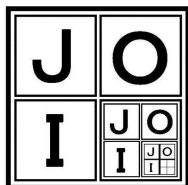
それに加えて、この施設では次の 2 つのことを行う。

まず、この施設には時おり「渋滞情報」が入ってくる。1 回の渋滞情報は 3 つの正の整数 r, s, t によって与えられる。これは「高速道路 r の上りの所要時間が s 、下りの所要時間が t である」ことを表す。ただし高速道路の上りとは、その高速道路の始点と終点の都市のうち番号が小さいほうの都市から大きいほうの都市へと向かう方向を指し、下りとはその逆方向を指すものとする。この渋滞情報に従って、施設のデータを更新する。

また、この施設には「問い合わせ」の電話がかかってくることもある。1 回の問い合わせは 2 つの正の整数 x, y によって与えられる。このとき、施設で管理している現在のデータをもとに「都市 x から都市 y への所要時間」を計算して答えなければならない。

課題 (TASK)

ある 1 日の間の「渋滞情報」と「問い合わせ」(これらをまとめてクエリという)の列が時刻順に与えたとき、それぞれの問い合わせごとに、その問い合わせの答えを出力するプログラムを作成せよ。ただし、最初の渋滞情報が入ってくる前の段階では、 $N-1$ 本すべての高速道路の所要時間は上下線ともに 1 である。



制限 (CONSTRAINTS)

- $2 \leq N \leq 100,000$ 都市の数
 $1 \leq M \leq 100,000$ 渋滞情報の数と問い合わせの数の和

入力 (INPUT)

標準入力から以下の入力を読み込め。

- 1行目には整数 N と M が空白を区切りとして書かれている。
- 続く $N - 1$ 行は、1行につき1本の高速道路について記述している。これらの行の i 行目には2つの整数 p_i, q_i ($1 \leq p_i < q_i \leq N$) が空白を区切りとして書かれており、これは高速道路 i が都市 p_i と都市 q_i を結んでいることを表す。
- 続く M 行は、1行につき1つのクエリ (渋滞情報または問い合わせ) について記述しており、次のどちらか一方が書いてある：
 - 渋滞情報…1文字 'I' と、整数 r ($1 \leq r \leq N - 1$), s ($1 \leq s \leq 1,000$), t ($1 \leq t \leq 1,000$)。それぞれは空白を区切りとして与えられる。
 - 問い合わせ…1文字 'Q' と、異なる2つの整数 x ($1 \leq x \leq N$), y ($1 \leq y \leq N$)。それぞれは空白を区切りとして与えられる。

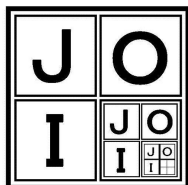
出力 (OUTPUT)

標準出力に以下のデータを出力せよ。

- 出力するデータの行数は入力中に現れる文字 'Q' の個数である。 i 行目は、第 i 番目の問い合わせの答えを表す1つの整数を含む。

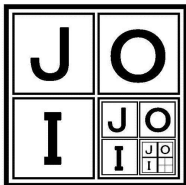
採点基準 (GRADING)

20点分のテストグループにおいて、 N, M は1,000を超えない。また、50点分のテストグループにおいて、任意の都市から別の任意の都市へ必ず1,000本以下の高速道路を乗り継いで移動することができる。



入出力例 (EXAMPLE)

入力例 (Sample Input)	出力例 (Sample Output)
4 5	2
1 3	13
3 4	12
2 3	
I 1 7 9	
Q 2 4	
I 3 12 11	
Q 2 4	
Q 4 2	



湖 (Lake)

カナダの南東部，アメリカとの国境の場所には「五大湖」として知られる有名な5つの湖がある．このたびカナダで IOI が開催されるのに伴い，会場にもっとも近いオンタリオ湖に観光船を運航する計画がいくつも持ち上がった．

おのこの観光船の計画は，湖の外周の2点を結ぶものであり，計画は全部で N 個ある． i 個目の計画は，地点 s_i と地点 t_i を結ぶ観光船を運航するというものである．ここで，地点 x とは，湖の東の端から周に沿って反時計回りに距離 x メートルだけ進んだ地点のことを指す．湖の一周の長さは 500,000 メートルである．

これらの中からできるだけ多くの計画を実現させたいが，船同士の衝突を避けるため，2つの航路が交差してはならない．

課題 (TASK)

N 個の運航計画が与えられたとき，実現できる計画の個数の最大値を求めるプログラムを作成せよ．

制限 (CONSTRAINTS)

$$1 \leq N \leq 2,000 \quad \text{計画の数}$$
$$0 \leq s_i < 500,000, 0 \leq t_i < 500,000 \quad \text{地点の座標}$$

入力 (INPUT)

標準入力から以下の入力を読み込め．

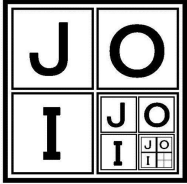
- 入力の1行目には整数 N が書かれている．これは，観光船を運航する計画の個数を表す．
- 入力の $i+1$ 行目 ($1 \leq i \leq N$) には2つの整数 s_i, t_i が空白を区切りとして書かれている．これらは， i 番目の計画で結ばれる予定である2つの地点を表す． $s_1, \dots, s_N, t_1, \dots, t_N$ の計 $2N$ 個の値はすべて異なる．

出力 (OUTPUT)

標準出力に，与えられた運航計画のうち，実現できる計画の個数の最大値を表す1つの整数を出力せよ．

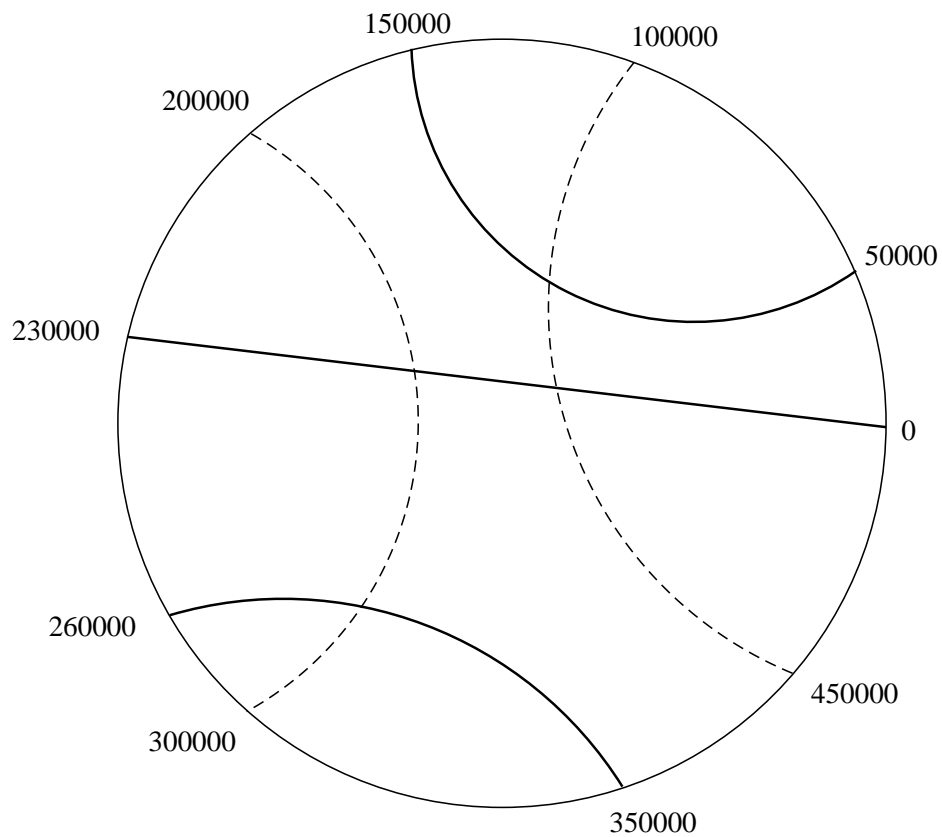
採点基準 (GRADING)

40点分のテストグループにおいて， $N \leq 200$ である．

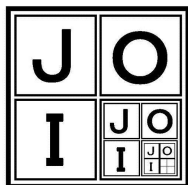


入出力例 (EXAMPLE)

入力例 (Sample Input)	出力例 (Sample Output)
5 50000 150000 450000 100000 200000 300000 260000 350000 0 230000	3



上の入力例における5つの計画を表した図(地点間の間隔は正確ではない)。太線の3つの計画を選べば航路が交差することなく船を運航できる。



プラグ (Plugs)

JOI 国には電気プラグを作っている会社が N 社あり、JOI 国は各社に 1 から N までの整数を ID として割り当てている。各電気プラグ会社は 1 組の電気プラグとソケットを作っているが、困ったことにすべての会社の電気プラグおよびソケットは他社のそれらとは違った形状をしている。

JOI 国の法律でソケットには電気プラグ会社の ID を印字することになっているのだが、電気プラグには ID は印字されていない。とある電気店の店長は、そのような JOI 国の事情もあって、顧客の要求にすばやく対応できるように、JOI 国に存在する N 種類すべての電気プラグを 1 つずつ会社の ID 順に入れた道具箱を持っていた。しかしある日、ふとした拍子に店長は道具箱の中身をバラバラにしてしまった。店長はある種のソケットにある種の電気プラグが入らないことは見抜けても、電気プラグを見てどの電気プラグ会社の電気プラグかを判断することはできないため、道具箱の中身を元に戻すことができなくなってしまった。店長は困り果ててしまった末、どのような困難な問題でもいとも簡単に解いてしまうことで名高い L 教授に解決を依頼した。

L 教授は順番がばらばらになった電気プラグに整理のため 1 から N までの番号をふり、それを元に店長から M 個の証言を引き出した。店長の k 個目の証言は「電気プラグ会社の ID が A_k から B_k のソケットには、 C_k 番目から D_k 番目のいずれの電気プラグも入らない」というものである。その後、L 教授は「謎は解決した。この M 個の証言を満たす電気プラグと会社の対応関係は 1 つに決まった。あとは君にまかせたよ。」と言って帰ってしまった。非常に理不尽な話であるが、弟子であるあなたは問題を解決し電気プラグと会社の対応関係を店長に知らせなければならない。

課題 (TASK)

店長の証言から電気プラグの対応関係を特定するプログラムを作成せよ。

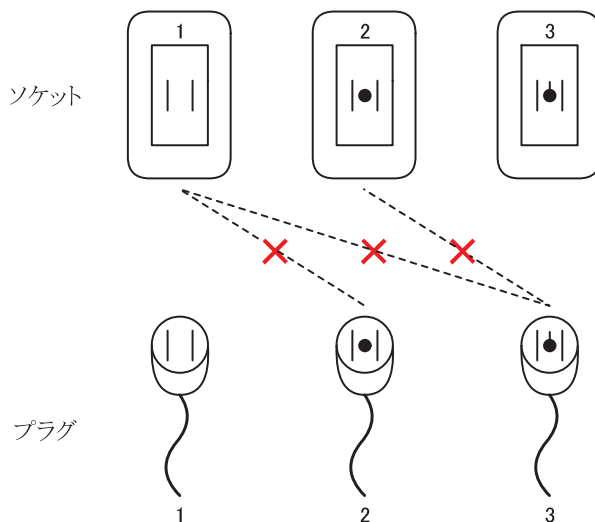
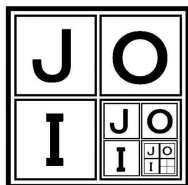


図 1



制限 (CONSTRAINTS)

- $1 \leq N \leq 3,000$ JOI 国に存在する電気プラグ会社の数
 $1 \leq M \leq 100,000$ 店長の証言の数
 $1 \leq A_k \leq B_k \leq N, 1 \leq C_k \leq D_k \leq N, 1 \leq k \leq M$ k 個目の証言の内容

入力 (INPUT)

標準入力から以下の入力を読み込め。

- 1 行目には整数 N と M が空白を区切りとして書かれている。
- 続く M 行は、1 行につき 1 つの証言について記述している。これらの行のうちの k 行目は k 番目の証言について記述しており、整数 A_k, B_k, C_k, D_k が空白を区切りとして書かれている。

出力 (OUTPUT)

標準出力に以下のデータを出力せよ。

- データは N 行からなり、 i 行目は ID が i の会社で作っている電気プラグの番号を含む。

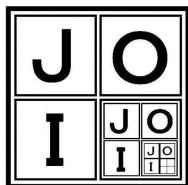
採点基準 (GRADING)

20 点分のテストグループにおいて N および M は 100 を超えない。

入出力例 (EXAMPLE)

入力例 (Sample Input)	出力例 (Sample Output)
3 2	1
1 1 2 3	2
1 2 3 3	3

上記の例は図 1 と一致する。図 1 の点線は店長の証言により入らないことがわかっている組み合わせである。



入力例 (Sample Input)	出力例 (Sample Output)
8 7	2
2 4 2 3	4
4 7 1 2	1
6 8 1 4	6
3 4 3 5	3
5 6 6 8	5
6 7 6 7	8
7 8 6 6	7