



BÀI A. BIỂU THỨC TOÁN HỌC

Timelimit: 1s

Cho dãy số $A[]$ gồm N phần tử và số nguyên dương M . Nhiệm vụ của bạn là đếm số cách điền các dấu $+$, $-$, $*$ vào giữa các phần tử $A[i]$ và $A[i+1]$ (với $i < N$) để tạo thành biểu thức có giá trị chia hết cho M .

Input:

Dòng đầu tiên là số lượng bộ test T ($T \leq 100$).

Mỗi test bắt đầu bởi 2 số nguyên N và M ($2 \leq N \leq 10$, $1 \leq M \leq 10^9$).

Dòng tiếp theo gồm N số nguyên $A[i]$ ($0 \leq A[i] \leq 10^9$).

Output:

Với mỗi test, hãy in ra đáp án tìm được trên một dòng.

Test ví dụ:

Input	Output
2	2
4 19	81
3 4 5 7	
5 1	
1 9 8 5 3	

Giải thích test 1:

$$3 + 4 + 5 + 7 = 19$$

$$3 * 4 - 5 - 7 = 0$$

BÀI B. SỐ MAY MẮN ĐẶC BIỆT

Time limit: 1s.

Trong một số nước châu Á, 8 và 6 được coi là những chữ số may mắn. Bất cứ số nguyên nào chỉ chứa chữ số 8 và 6 được coi là số may mắn, ví dụ 6, 8, 66, 668, 88, 886 ...

Số may mắn đặc biệt là các số may mắn có dạng $S = 8...86...6$ (các chữ số 8 liên tiếp nhau, rồi các chữ số 6 liên tiếp nhau). Trong đó S phải có ít nhất một chữ số, và chữ số 6 và 8 không nhất thiết phải đồng thời xuất hiện. Ví dụ, 8, 88, 6, 66, 86, 886, 8866 ... là các số có dạng S.

Cho trước một số nguyên dương X, hãy tìm số may mắn đặc biệt nhỏ nhất có không quá 200 chữ số và chia hết cho X.

Input:

Dòng đầu tiên là số lượng bộ test T ($1 \leq T \leq 100$).

Mỗi test gồm một số nguyên dương X duy nhất ($1 < X < 10000$).

Output:

Với mỗi test, in ra trên một dòng số may mắn đặc biệt S nhỏ nhất chia hết cho X. Trường hợp không tồn tại số S có không quá 200 chữ số như vậy, ghi -1.

Test ví dụ:

Input	Output
4	6
6	8
8	86
43	-1
5	

BÀI C. TÍCH CỦA DÃY SỐ

Time limit: 1s.

Cho hai số nguyên dương N và M . Hãy đếm xem có bao nhiêu dãy số $A[]$ có N phần tử nguyên dương thỏa mãn $A[1] \times A[2] \times \dots \times A[N] = M$.

Hai dãy số $A[]$ và $B[]$ được gọi là khác nhau nếu như tồn tại phần tử i nào đó sao cho $A[i] \neq B[i]$.

Input:

Gồm 2 số nguyên dương N và M ($1 \leq N \leq 10^5$, $1 \leq M \leq 10^9$).

Output:

In ra số lượng dãy số phân biệt thỏa mãn yêu cầu của đề bài. Vì đáp số có thể rất lớn, hãy in ra đáp số theo modulo 10^9+7 .

Test ví dụ:

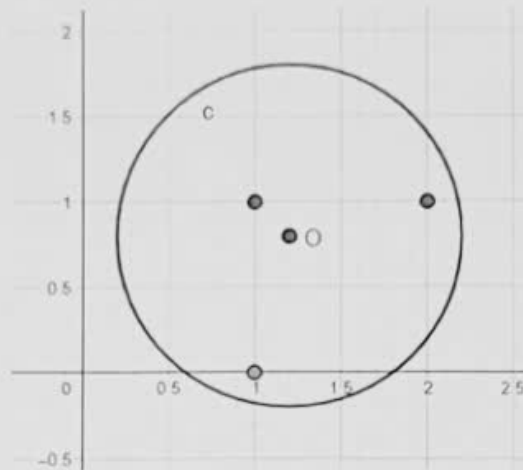
Input	Output
2 6	4
3 10	9
45 426	91125

Giải thích test 1: Có 4 dãy số thỏa mãn là $(1, 6)$, $(2, 3)$, $(3, 2)$, $(6, 1)$.

BÀI D. ĐƯỜNG TRÒN

Time limit: 1s.

Cho đường tròn có tâm tại vị trí (X, Y) và bán kính R . Hãy đếm số lượng các điểm có tọa độ nguyên nằm bên trong hoặc trên đường tròn?



Input:

Gồm 3 số thực X, Y, R có tối đa 4 chữ số sau dấu phẩy ($|X|, |Y| \leq 10^5, 0 \leq R \leq 10^5$).

Output:

In ra số lượng điểm có tọa độ nguyên nằm bên trong hoặc bên trên đường tròn đã cho.

Test ví dụ:

Input	Output
1.2 0.8 1	3
0 0 1	5
-39066.9606 -83906 45886.5164	6614851027

BÀI E. NỐI ĐIỂM

Time limit: 1s.

Cho N điểm trên mặt phẳng Oxy.

Để vẽ được đoạn thẳng nối A và B sẽ tốn chi phí bằng $(x_A - x_B)^2 + (y_A - y_B)^2$. Nhiệm vụ của bạn là hãy nối các điểm lại với nhau, sao cho N điểm đã cho tạo thành 1 thành phần liên thông duy nhất và chi phí thực hiện là nhỏ nhất có thể.

Input:

Mỗi test bắt đầu bởi số nguyên N ($1 \leq N \leq 100000$).

N dòng tiếp theo, mỗi dòng gồm 2 số nguyên $x[i]$, $y[i]$ là tọa độ của điểm thứ i .

Giới hạn: $0 \leq x[i] \leq 10^6$, $0 \leq y[i] \leq 10$.

Output:

In ra một số nguyên là chi phí nhỏ nhất tìm được.

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

gk

Test ví dụ:

Input	Output
4 1 1 2 2 2 4 6 2	22
10 50 9 77 3 86 5 50 1 49 10 30 0 90 3 12 2 82 10 40 4	1487

Giải thích test 1:

Nối điểm (1, 2) chi phí bằng 2.

Nối điểm (2, 3) chi phí bằng 4

Nối điểm (2, 4) chi phí bằng 16.

8

BÀI F. CHIA HẾT CHO 8

Time limit: 1s.

Cho một chuỗi S chỉ gồm các chữ số từ '1' tới '9'. Bạn được phép thực hiện vô hạn phép đổi chỗ các ký tự của chuỗi S với nhau. Nhiệm vụ của bạn là hãy xác định xem chuỗi mới thu có phải là một số nguyên chia hết chia 8 hay không?

Input:

Một chuỗi mô tả số S có độ dài không quá 200000 ký tự.

Output:

In ra "YES" nếu chuỗi mới thu được chia hết cho 8, in ra "NO" trong trường hợp ngược lại.

Test ví dụ:

Input	Output
1234	YES
1331	NO
61	YES
588	NO

BÀI G. TRUY VẤN TRÊN DÃY SỐ

Time limit: 2s.

Cho dãy số $A[]$ có N phần tử và M truy vấn, mỗi truy vấn thuộc một trong bốn loại sau:

I pos X: Chèn thêm phần tử X vào sau phần tử thứ pos của dãy số $A[]$ hiện tại.

U pos len Y: Thay đổi giá trị các phần tử $A[pos]$, $A[pos+1]$, ..., $A[pos + len - 1]$ thành giá trị Y .

R pos len: Xoay ngược đoạn các phần tử $A[pos]$, $A[pos+1]$, ..., $A[pos + len - 1]$.

Q pos len: Yêu cầu tính tổng các phần tử của đoạn $A[pos]$, $A[pos+1]$, ..., $A[pos + len - 1]$.

Input:

Dòng đầu tiên gồm 2 số nguyên N và M ($N, M \leq 200000$).

Dòng tiếp theo gồm N số nguyên mô tả dãy số $A[]$.

M dòng tiếp theo, mỗi dòng mô tả một truy vấn. Mỗi truy vấn sẽ nhập một kí tự đầu tiên trước.

Nếu là kí tự 'I', đây là truy vấn loại 1. Theo sau là 2 giá trị pos và X .

Nếu là kí tự 'U', đây là truy vấn loại 2. Theo sau là 3 giá trị pos , len và Y .

Nếu là kí tự 'R', đây là truy vấn loại 3. Theo sau là 2 giá trị pos và len .

Nếu là kí tự 'Q', đây là truy vấn loại 4. Theo sau là 2 giá trị pos và len .

Các phần tử $A[i]$ và X, Y có giá trị tuyệt đối không vượt quá 10^6 . Input đảm bảo các truy vấn loại 2, 3, 4 hợp lệ, tức $pos + len - 1$ không vượt quá độ dài của dãy số hiện tại.

Output:

Với mỗi truy vấn loại 4, hãy in ra tổng của dãy số tìm được trên một dòng.

Test ví dụ:

Input	Output
8 7	16
3 -5 1 4 -2 5 6 7	-10
Q 5 4	-3
I 1 -6	10
Q 2 3	
U 4 2 8	
Q 2 3	
R 3 3	
Q 2 3	

Giải thích test:

Truy vấn 1: $\text{Sum}[5 \rightarrow 8] = -2 + 5 + 6 + 7 = 16$

Sau truy vấn 2: Dãy số A[] mới là 3 -6 -5 1 4 -2 5 6 7

Truy vấn 3: $(-6) + (-5) + 1 = -10$

Sau truy vấn 4: Dãy số A[] mới là 3 -6 -5 8 8 -2 5 6 7

Truy vấn 5: $(-6) + (-5) + 8 = -3$

Sau truy vấn 6: Dãy số A[] mới là 3 -6 8 8 -5 -2 5 6 7

Truy vấn 7: $(-6) + 8 + 8 = 10$

BÀI H. DI CHUYỂN QUÂN XE

Time limit: 8s.

Luật chơi cờ vua đã rất quen thuộc. Trên bàn cờ mở rộng với N hàng và M cột đã đặt một quân Xe trắng và một số quân Mã đen. Mỗi bước, quân Xe có thể di chuyển tới các ô trống mà không bị quân Mã chiếm giữ (nếu không, quân Xe sẽ bị ăn), hoặc có thể ăn luôn một số quân Mã trên đường đi của nó nếu như vị trí này không được bảo vệ bởi một quân Mã khác.

Biết rằng các quân Mã sẽ không di chuyển và chỉ bảo vệ các vị trí mà chúng nắm giữ. Hãy kiểm tra xem có thể di chuyển quân Xe một cách an toàn đến vị trí đích xác định trước hay không?



Input:

Dòng đầu ghi hai số N và M là kích thước bàn cờ ($2 \leq N, M \leq 750$).

Tiếp theo là N dòng mô tả bàn cờ. Trong đó có đúng 1 chữ X là vị trí quân Xe, có đúng 1 chữ O là vị trí đích cần đến. Và có ít nhất 1 chữ M là vị trí các quân Mã. Các dấu $.$ là các ô trống.

Vị trí đích đến (chữ O) cũng có thể đã bị chiếm giữ bởi các quân Mã đen, khi đó tất nhiên quân Xe trắng sẽ phải ăn các quân Mã đó trước khi muốn di chuyển vào vị trí đích.

Output:

In ra "YES" nếu quân xe di được tới đích, in ra "NO" trong trường hợp ngược lại.

Test ví dụ:

Input	Output
6 6O ..M.M. M.M...M. X..M..M.	YES
6 6 ..M... ..MOM.MM.. ...M.. X.....	YES
4 4 .M.. MX.. M... .M.O	NO
4 3 XM. MM.O	YES

BÀI 1. XẾP HÀNG

Time limit: 1s.

Có N sinh viên xếp thành một hàng dọc, mỗi bạn được gán một con số từ 1 đến N. Mỗi sinh viên có lựa chọn duy nhất là nếu quyết định di chuyển thì sẽ được chuyển lên đầu hàng, những người phía trên sẽ lùi xuống một bước để nhường chỗ cho anh ta.

Cho thứ tự xếp hàng tại hai thời điểm t1 và t2 ($t1 < t2$). Hãy xác định xem trong khoảng thời gian này thì ít nhất bao nhiêu sinh viên đã quyết định di chuyển.

Input:

Dòng đầu ghi số bộ test T ($T \leq 10.000$).

Mỗi bộ test bắt đầu với một dòng ghi số nguyên dương N (không quá 100.000).

Dòng tiếp theo ghi N số là thứ tự tại thời điểm t1.

Dòng tiếp theo ghi N số là thứ tự tại thời điểm t2.

Output:

Với mỗi test, ghi ra giá trị kết quả tính được trên 1 dòng.

Test ví dụ:

Input	Output
2	2
5	4
1 4 2 5 3	
4 5 1 2 3	
8	
8 2 4 7 1 6 5 3	
8 5 6 1 4 8 2 7 3	

Giải thích test 1: Sinh viên 5 rồi sinh viên 4 thực hiện di chuyển.

8 5 6 1 4 7 2 3 }
8 2 4 7 1 6 5 3 }
8 5 6 1 4 7 2 3 }

$1, 2, 5$ &
 $2, 1, 5$
 $5, 1, 2$ } = 6

6 + 6 = 12

12 + 12 = 24

24 + 24 = 48

48 + 48 = 96

96 + 96 = 192

192 + 192 = 384

384 + 384 = 768

768 + 768 = 1536

1536 + 1536 = 3072

3072 + 3072 = 6144

6144 + 6144 = 12288

12288 + 12288 = 24576

24576 + 24576 = 49152

49152 + 49152 = 98304

98304 + 98304 = 196608

196608 + 196608 = 393216

393216 + 393216 = 786432

786432 + 786432 = 1572864

1572864 + 1572864 = 3145728

3145728 + 3145728 = 6291456

6291456 + 6291456 = 12582912

12582912 + 12582912 = 25165824

25165824 + 25165824 = 50331648

50331648 + 50331648 = 100663296

100663296 + 100663296 = 201326592

201326592 + 201326592 = 402653184

402653184 + 402653184 = 805306368

805306368 + 805306368 = 1610612736

1610612736 + 1610612736 = 3221225472

3221225472 + 3221225472 = 6442450944

6442450944 + 6442450944 = 12884901888

12884901888 + 12884901888 = 25769803776

25769803776 + 25769803776 = 51539607552

51539607552 + 51539607552 = 103079215104

103079215104 + 103079215104 = 206158430208

206158430208 + 206158430208 = 412316860416

412316860416 + 412316860416 = 824633720832

824633720832 + 824633720832 = 1649267441664

1649267441664 + 1649267441664 = 3298534883328

3298534883328 + 3298534883328 = 6597069766656

6597069766656 + 6597069766656 = 13194139533312

13194139533312 + 13194139533312 = 26388279066624

26388279066624 + 26388279066624 = 52776558133248

52776558133248 + 52776558133248 = 105553116266496

105553116266496 + 105553116266496 = 211106232532992

211106232532992 + 211106232532992 = 422212465065984

422212465065984 + 422212465065984 = 844424930131968

844424930131968 + 844424930131968 = 1688849860263936

1688849860263936 + 1688849860263936 = 3377699720527872

3377699720527872 + 3377699720527872 = 6755399441055744

6755399441055744 + 6755399441055744 = 13510798882111488

13510798882111488 + 13510798882111488 = 27021597764222976

27021597764222976 + 27021597764222976 = 54043195528445952

54043195528445952 + 54043195528445952 = 108086391056891904

108086391056891904 + 108086391056891904 = 216172782113783808

216172782113783808 + 216172782113783808 = 432345564227567616

432345564227567616 + 432345564227567616 = 864691128455135232

864691128455135232 + 864691128455135232 = 1729382256910270464

1729382256910270464 + 1729382256910270464 = 3458764513820540928

3458764513820540928 + 3458764513820540928 = 6917529027641081856

6917529027641081856 + 6917529027641081856 = 13835058055282163712

13835058055282163712 + 13835058055282163712 = 27670116110564327424

27670116110564327424 + 27670116110564327424 = 55340232221128654848

55340232221128654848 + 55340232221128654848 = 110680464442257309696

110680464442257309696 + 110680464442257309696 = 221360928884514619392

221360928884514619392 + 221360928884514619392 = 442721857769029238784

442721857769029238784 + 442721857769029238784 = 885443715538058477568

885443715538058477568 + 885443715538058477568 = 1770887431076116955136

1770887431076116955136 + 1770887431076116955136 = 3541774862152233910272

3541774862152233910272 + 3541774862152233910272 = 7083549724304467820544

7083549724304467820544 + 7083549724304467820544 = 14167099448608935641088

14167099448608935641088 + 14167099448608935641088 = 28334198897217871282176

28334198897217871282176 + 28334198897217871282176 = 56668397794435742564352

56668397794435742564352 + 56668397794435742564352 = 113336795588871485128704

113336795588871485128704 + 113336795588871485128704 = 226673591177742970257408

226673591177742970257408 + 226673591177742970257408 = 453347182355485940514816

Khi quan sát các từ khóa “ICPC”, “PTIT”, “2024” ta thấy chúng có một đặc điểm chung là đều có độ dài là 4, chứa 3 kí tự phân biệt, giữa 2 kí tự giống nhau là 1 kí tự khác, ta tạm gọi đặc điểm này là “4321”.

- Có độ dài là 4, chứa 3 kí tự phân biệt, giữa 2 kí tự giống nhau là 1 kí tự khác
- Nếu tồn tại hai xâu A và B cùng có tính chất “4321” thì khi ta ghép hai xâu A, B với nhau ta được một xâu mới cũng có tính chất “4321”, ví dụ: “20242024”, “ICPCPTIT2024”

Xâu A được xem là xâu con của xâu B nếu ta có thể xóa không hoặc một vài kí tự khỏi xâu B sao cho các kí tự còn lại tạo thành xâu A. Ví dụ xâu “ICT” có tất cả 7 xâu con {“I”, “C”, “T”, “IC”, “CT”, “IT”, ICT}.

Dòng đầu chứa số bộ test T ($T \leq 5$)

Output:

Dòng đầu tiên in ra số cách chọn xâu con có tính chất “4321” chia dư cho 10^9+7 .

14

Test ví dụ:

Input	Output
3	96
ICPCPTIT2024	ICPCPTIT2024
ITPTIT	5
20I24	IPIT
	2
	2024

BÀI K. XÁC ĐỊNH CHỮ SỐ

Time limit: 1s.

Viết liên tiếp các số tự nhiên từ 1 đến 10^{15} ta nhận được một số nguyên dương N . Tiến hành xóa đi k chữ số của số N để được số lớn nhất M . Người ta muốn biết chữ số thứ p của M là chữ số nào.

Yêu cầu: Cho hai số nguyên dương k, p , hãy tìm chữ số thứ p của số M theo cách tạo trên.

Input

- Dòng đầu chứa số nguyên T là số bộ dữ liệu;
- T dòng sau, mỗi dòng chứa hai số nguyên dương k, p ($k, p \leq 10^{15}$).

Output

- Gồm T dòng, mỗi dòng chứa một số nguyên là kết quả tương ứng của bộ test trong dữ liệu vào.

Test ví dụ

Input	Output
2	8
5 3	9
9 1	

BÀI L VỊ TRÍ ĐẶT GƯƠNG

Time limit: 1s.

Mặt sàn phòng làm việc của Alice được mô tả bằng một đa giác lồi n đỉnh với n bức tường tương ứng là các cạnh của đa giác nối hai đỉnh liền kề. Các đỉnh được đánh số từ 1 đến n ngược chiều kim đồng hồ, đỉnh thứ i ($1 \leq i \leq n$) có tọa độ là (x_i, y_i) . Các cạnh được đánh số từ 1 đến n theo thứ tự ngược chiều kim đồng hồ bắt đầu từ cạnh thứ nhất nối giữa đỉnh số 1 và số 2. Vị trí bàn làm việc của Alice là (u, v) nằm trong đa giác. Alice dự định thiết kế và đặt một tấm gương chiếm phủ kín một bức tường nào đó trong n bức tường để tổng diện tích mặt cắt mà Alice có thể quan sát được từ trong gương khi đứng ở tọa độ (u, v) hướng mặt về gương là lớn nhất.

Yêu cầu: Hãy giúp Alice chọn bức tường để đặt gương mà tổng diện tích mặt cắt Alice có thể quan sát được từ trong gương khi đứng ở tọa độ (u, v) hướng mặt về gương là lớn nhất.

Input:

- Dòng đầu chứa số nguyên dương n ($3 \leq n \leq 10^5$) là số đỉnh của đa giác lồi;
- Trong n dòng tiếp, dòng thứ i ($1 \leq i \leq n$) chứa hai số nguyên x_i, y_i ($|x_i|, |y_i| \leq 10^8$), mô tả các đỉnh của đa giác theo thứ tự ngược chiều kim đồng hồ;
- Dòng cuối chứa hai số nguyên u, v là tọa độ bàn làm việc của Alice (nằm hoàn toàn bên trong đa giác).

Output:

- In ra một số thực là tổng diện tích lớn nhất (với độ chính xác 10^{-5}) của mặt cắt mà Alice có thể quan sát được từ trong gương khi đứng ở tọa độ (u, v) hướng mặt về gương đặt ở bức tường tối ưu đã được lựa chọn.

Test ví dụ:

Input	Output	Minh họa
4 10 5 30 5 40 20 20 20 30 10	300.00000	<p>Diện tích quan sát được tương ứng các vị trí đặt gương ở các bức tường từ 1 đến 4 là: 225.00000, 300.00000, 300.00000, 235.38461.</p>