**LUYỆN TẬP CÁC CẤU TRÚC DỮ LIỆU NÂNG CAO**

**PHẦN 1. BST**

BÀI 1. MUA TRANG SỨC

Một cửa hàng trang sức có n món hàng, trang sức thứ i có vẻ đẹp bằng b[i] và giá bán là p[i]. Tuy nhiên, cửa hàng này chỉ bán hàng cho người nước ngoài, và các giao dịch được thực hiện bằng bảng Anh hoặc đô la. Các trang sức được bán đã niêm yết giá theo 1 trong 2 loại tiền tệ trên.

Luna là một khách hàng lần đầu tới cửa hàng. Cô có X đồng bảng Anh và Y đồng đô la. Cô muốn mua 2 món hàng sao cho chỉ số vẻ đẹp của chúng là lớn nhất. Các bạn hãy xác định xem giá trị vẻ đẹp lớn nhất của 2 món hàng mà Luna mua được bằng bao nhiêu. Nếu Luna không đủ tiền để mua 2 món hàng, in ra 0.

Lưu ý, không được phép thực hiện đổi tiền giữa đồng bảng Anh và đô la.

**Input:**

Dòng đầu tiên gồm 3 số nguyên N, X, Y (2 ≤ N ≤ 100 000, 0 ≤ X, Y ≤ 100 000).

N dòng tiếp theo, mỗi dòng gồm b[i], p[i] (1 ≤ b[i], p[i] ≤ 100 000) và 1 kí tự, lần lượt là vẻ đẹp và giá của món hàng thứ i. Kí tự trên bằng ‘C’ có nghĩa món hàng được niêm yết theo đồng bảng Anh, ‘D’ có nghĩa được niêm yết theo đồng đô la.

**Output:**

In ra một số nguyên là đáp án của bài toán

**Test ví dụ:**

|  |  |  |
| --- | --- | --- |
| Test 1 | Test 2 | Test 3 |
| Input:  3 7 6  10 8 C  4 3 C  5 6 D  Output:  9 | Input:  2 4 5  2 5 C  2 1 D  Output:  0 | Input:  3 10 10  5 5 C  5 5 C  10 11 D  Output:  10 |

Giải thích test 3: Luna mua 2 món hàng đầu tiên bằng bảng Anh.

BÀI 2. MUA ÁO

Một cửa thời trang có N chiếc áo được đặc trưng bởi các tham số p[i], a[i], b[i], trong đó p[i] là giá bán, a[i] là màu mặt trước của chiếc áo, b[i] là màu ở mặt sau. Các giá trị p[i] là khác nhau, không tồn tại 2 chiếc áo nào có cùng giá bán.

Có M khách hàng lần lượt vào mua hàng. Khách hàng thứ i thích màu c[i] và họ sẽ mua 1 chiếc áo nếu như có màu họ yêu thích ở phía trước hoặc phía sau. Dĩ nhiên, khách hàng luôn chọn chiếc áo có giá rẻ nhất.

Nhiệm vụ của bạn là hãy xác định giá chiếc áo mua được đối với mỗi khách hàng. Nếu khách hàng nào không mua được áo, in ra -1.

**Input:**

Dòng đầu tiên gồm số nguyên N (1 ≤ N ≤ 200 000).

3 dòng tiếp theo, mỗi dòng gồm N số nguyên, lần lượt là p[1], p[2], …, p[N]; a[1], a[2], …, a[N] và b[1], b[2], …, b[N].

Tiếp theo là số nguyên M (1 ≤ M ≤ 200 000).

Dòng cuối gồm M số nguyên c[i]. (1 ≤ a[i], b[i], c[i] ≤ 3).

**Output:**

In ra M số nguyên tìm được trên một dòng.

**Test ví dụ:**

|  |  |
| --- | --- |
| Test 1 | Test 2 |
| Input:  5  300 200 400 500 911  1 2 1 2 3  2 1 3 2 1  6  2 3 1 2 1 1  Output:  200 400 300 500 911 -1 | Input:  2  1000000000 1  1 1  1 2  2  2 1  Output:  1 1000000000 |

BÀI 3. TẬP LỆNH VỚI FILE

Tý đang viết một chương trình quản lý file làm bài tập lớn môn hệ điều hành. Các tập lệnh như sau:

* Tạo file: “crt file\_name”

1. Tạo ra file có tên file\_name.
2. Nếu file\_name đã tồn tại, thêm chỉ số (x) vào đằng sau tên file trong đó x là số dương nhỏ nhất chưa xuất hiện. Chẳng hạn crt aa sẽ tạo ra file aa(2) nếu như file aa, aa(1) đã tồn tại.
3. In ra “+ new\_file” trên một dòng, trong đó new\_file là tên của file mới.

* Xóa file: “del file\_name”

1. Xóa file có tên là file\_name. File này chắc chắn có tồn tại.
2. In ra “- file\_name”, trong đó file\_name là tên file bị xóa.

* Đổi tên file: “rnm file\_name1 file\_nam2”

1. Đổi tên file\_name1 thành file\_name2 (file\_nam1 chắc chắn có tồn tại). Thao tác này tương đương với việc xóa file\_name1 và tạo ra file mới file\_name2, file\_name2 phải tuân theo quy tắc đặt tên của lệnh tạo file.
2. In ra “r file\_name1 -> new\_file”, trong đó new\_file là tên của file mới.

Cho biết trước tập lệnh đã nhập, các bạn hãy thử chạy chương trình của Tý xem kết quả xuất ra màn hình như thế nào?

**Input:**

Dòng đầu tiên là số lượng lệnh Q (1 ≤ Q ≤ 100 000).

Q dòng tiếp theo, mỗi dòng gồm một lệnh như mô tả ở trên. Tên file chỉ chứa các chữ cái tiếng anh thường.

**Output:**

In ra Q dòng là kết quả của mỗi lệnh trên một dòng.

**Test ví dụ:**

|  |  |
| --- | --- |
| Input: | Output |
| 11  crt phonebook  crt phonebook  crt phonebook  crt todo  crt phonebook  del phonebook  del phonebook(2)  crt phonebook  crt phonebook  crt phonebook  rnm phonebook(2) todo | + phonebook  + phonebook(1)  + phonebook(2)  + todo  + phonebook(3)  - phonebook  - phonebook(2)  + phonebook  + phonebook(2)  + phonebook(4)  r phonebook(2) -> todo(1) |

Giải thích test:

Tập hợp các file sau mỗi lệnh:

(1) phonebook

(2) phonebook, phonebook(1)

(3) phonebook, phonebook(1), phonebook(2)

(4) phonebook, phonebook(1), phonebook(2), todo

(5) phonebook, phonebook(1), phonebook(2), todo, phonebook(3)

(6) phonebook(1), phonebook(2), todo, phonebook(3)

(7) phonebook(1), todo, phonebook(3)

(8) phonebook(1), todo, phonebook(3), phonebook

(9) phonebook(1), todo, phonebook(3), phonebook, phonebook(2)

(10) phonebook(1), todo, phonebook(3), phonebook, phonebook(2), phonebook(4)

(11) phonebook(1), todo, phonebook(3), phonebook, todo(1), phonebook(4)

BÀI 4. ĐƯỜNG ĐUA

Lần này nông dân Bell quyết định tổ chức cuộc đua một cách đồng thời cho tất cả các con bò của mình. Các con bò có vị trí xuất phát khác nhau và một trong số chúng có tốc độ khác nhau.

Do có con chạy nhanh, con chạy chậm, nên nếu xếp chúng vào cùng một làn chạy, sẽ có lúc nào đó chúng sẽ dẫm đạp lên nhau. Vì vậy, anh Bell phải tách chúng ra và xếp vào các làn chạy khác nhau.

Cuộc thi chạy lần này diễn ra trong T phút. Các bạn hãy tính giúp anh Bell xem cần phải chuẩn bị ít nhất bao nhiêu làn đua cho cuộc thi lần này.

**Input:**

* Dòng đầu tiên gồm hai số nguyên N và T (1 ≤ N ≤ 10^5, 1 ≤ T ≤ 10^9).
* N dòng tiếp theo, mỗi dòng gồm 2 số nguyên cho biết vị trí xuất phát P[i] và tốc độ V[i] của thí sinh thứ i (1 ≤ P[i], V[i] ≤ 10^9). Dữ liệu được sắp xếp theo thứ tự tăng dần vị trí xuất phát của các thí sinh.

**Output:**

* In ra một số nguyên là đáp án của bài toán.

**Test ví dụ:**

|  |  |
| --- | --- |
| Input | Output |
| 5 3  0 1  1 2  2 3  3 2  6 1 | 3 |

**Giải thích test 1:**

Lane 1: Gồm con bò thứ 1 và 5.

Lane 2: Gồm con bò thứ 2 và 4.

Lane 3: Gồm con bò thứ 3.

BÀI 5. CÂY NHỊ PHÂN TÌM KIẾM

Nhiệm vụ của bạn là xây dựng một cây tìm kiếm nhị phân, rồi lần lượt thêm các phần tử trong dãy số A[] theo đúng thứ tự.

Tiếp theo, hãy sử dụng phép DFS từ gốc của cây và in ra thứ tự DFS.

**Input:**

* Dòng đầu tiên là số lượng bộ test T (T ≤ 20).
* Mỗi test bắt đầu bởi số nguyên N (1 ≤ N ≤ 1000).
* Dòng tiếp theo gồm N số nguyên A[i] (0 ≤ A[i] ≤ 5000), là các số lần lượt được thêm vào cây. Dữ liệu đảm bảo các số A[i] là phân biệt.

**Output:**

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

**Test ví dụ:**

|  |  |
| --- | --- |
| Input | Output |
| 2  7  50 30 20 40 70 60 80  10  10 9 8 7 6 5 4 3 2 1 | 50 30 20 40 70 60 80  10 9 8 7 6 5 4 3 2 1 |

Giải thích test 1:

              50

           /     \

          30      70

         /  \    /  \

       20   40  60   80

**PHẦN 2. IT**

BÀI 6. GIÁ TRỊ LỚN NHẤT – v1

Cho dãy số gồm N phần tử có giá trị ban đầu bằng 0.

Có M phép biến đổi, mỗi phép có dạng (u, v, k): tăng mỗi phần tử từ u tới v lên k đơn vị.

Cho Q câu hỏi, mỗi câu hỏi có dạng (u, v): yêu cầu tìm phần tử lớn nhất trong đoạn [u, v].

**Input**

* Dòng đầu tiên gồm 2 số nguyên N và M (1 ≤ N, M ≤ 100 000).
* M dòng tiếp theo, mỗi dòng gồm 3 số u, v, k (1 ≤ k ≤ 10 000).
* Dòng tiếp theo là số nguyên Q (1 ≤ Q ≤ 100 000).
* Q dòng tiếp theo, mỗi dòng gồm 2 số nguyên u, v.

**Output**

* Với mỗi truy vấn, hãy in ra đáp án trên một dòng.

**Test ví dụ:**

|  |  |
| --- | --- |
| Input | Output |
| 6 2  1 3 2  4 6 3  1  3 4 | 3 |

BÀI 7. GIÁ TRỊ LỚN NHẤT – v2

Cho dãy số gồm N phần tử có giá trị ban đầu bằng 0.

Có Q truy vấn:

Loại 1 có dạng (1, u, v, k): tăng mỗi phần tử từ u tới v lên k đơn vị (1 ≤ k ≤ 10 000).

Loại 2 có dạng (2, u, v): yêu cầu tìm phần tử lớn nhất trong đoạn [u, v].

**Input**

* Dòng đầu tiên gồm 2 số nguyên N và Q (1 ≤ N, M ≤ 100 000).
* Q dòng tiếp theo, mỗi dòng gồm một trong hai loại truy vấn như trên.

**Output**

* Với mỗi truy vấn loại 2, hãy in ra đáp án trên một dòng.

**Test ví dụ:**

|  |  |
| --- | --- |
| Input | Output |
| 6 3  1 1 3 3  1 4 6 4  2 1 6 | 4 |

BÀI 8. DÃY CON NHỎ NHẤT CÓ UCLN BẰNG K

Cho dãy số A[] có N phần tử. Nhiệm vụ của bạn là tìm dãy con liên tiếp có độ dài nhỏ nhất, sao cho UCLN của các phần tử đúng bằng K.

**Input**

* Dòng đầu tiên là số lượng bộ test T (T ≤ 10).
* Mỗi test bắt đầu bằng 2 số nguyên N và K (1 ≤ N ≤ 100 000).
* Dòng tiếp theo gồm N số nguyên A[i] (1 ≤ A[i], K ≤ 10^9).

**Output**

* Với mỗi test, hãy in ra đáp án trên một dòng. Nếu không tìm được dãy con nào, in ra -1.

**Test ví dụ:**

|  |  |
| --- | --- |
| Input | Output |
| 3  8 3  6 9 7 10 12 24 36 27  4 3  2 4 6 8  4 6  1 2 3 6 | 2  -1  1 |

BÀI 9. PHẦN TỬ TRUNG VỊ

Phần tử trung vị của một dãy số có n phần tử là phần tử chính giữa (số thứ a[(n+1)/2]) sau khi dãy số được sắp xếp. Ví dụ dãy A[] = [1, 2, 3, 4, 5], phần tử trung vị là 3.

Cho một dãy số ban đầu rỗng và Q truy vấn, mỗi truy vấn có dạng:

* add x: Thêm số x vào dãy số (1 ≤ x ≤ 10^6).
* del x: Xóa bỏ x (x đảm bảo tồn tại trong dãy)
* print: Yêu cầu in ra phần tử trung vị của dãy số.

**Input**

* Dòng đầu tiên là số lượng truy vấn Q (Q ≤ 100 000).
* Q dòng tiếp theo, mỗi dòng gồm một trong 3 loại truy vấn như trên.

**Output**

* Với mỗi truy vấn “print”, hãy in ra phần tử trung vị của dãy số.

**Test ví dụ:**

|  |  |
| --- | --- |
| Input | Output |
| 10  add 1  add 4  add 7  add 8  print  add 9  print  del 1  del 8  print | 4  7  7 |

Giải thích truy vấn 1: Dãy số hiện tại là [1, 4, 7, 8]. Phần tử trung vị là 4.

**PHẦN 3. BIT**

BÀI 10. K-QUERY

Cho dãy số A[] có N phần tử. Có Q truy vấn dạng (u, v, K) yêu cầu bạn xác định số lượng phần tử lớn hơn K trong dãy con A[u], A[u+1], ..., A[v].

**Input**

* Dòng đầu tiên là số nguyên N (1 ≤ N ≤ 100 000).
* Dòng tiếp theo gồm N số nguyên A[i] (1 ≤ A[i] ≤ 10^9).
* Tiếp theo là số lượng truy vấn Q (1 ≤ Q ≤ 100 000).
* Q dòng tiếp theo, mỗi dòng gồm 3 số nguyên u, v, K (1 ≤ u, v ≤ N, 1 ≤ K ≤ 10^9).

**Output**

* Với mỗi truy vấn, in ra đáp án tìm được trên một dòng.

**Test ví dụ:**

|  |  |
| --- | --- |
| Input | Output |
| 5  5 1 2 3 4  3  2 4 1  4 4 4  1 5 2 | 2  0  3 |

BÀI 11. CUỘC ĐUA KỲ THÚ

Với một trang trại rộng lớn, nông dân Bell quyết định tổ chức của đua cho những con bò của mình để nâng cao thể lực. Hành trình của cuộc đua gồm có N trạm. Mỗi thí sinh có một hành trình khác nhau và phải tuân thủ chạy lần lượt theo thứ tự từ trạm xuất phát đến trạm đích.

Để tăng thêm sự thú vị cho cuộc thi, thỉnh thoảng anh Bell sẽ thay đổi vị trí của các trạm. Biết rằng các con bò của mình không đủ thể lực, anh Bell cho phép các thí sinh được bỏ qua một trạm nào đó trong hành trình của nó. Và anh vẫn thầm nghĩ rằng, chắc chúng chẳng chọn ra được giải pháp tối ưu đâu.

Các con bò của nông dân Bell không được thông minh cho lắm, các bạn hãy giúp chúng tính quãng đường ngắn nhất để hoàn thành chặng đua của mình.

Khoảng cách giữa 2 trạm có tọa độ (x1, y1) và (x2, y2) được tính bằng khoảng cách Manhattan theo công thức |x1-x2| + |y1-y2|.

**Input**

* Dòng đầu tiên gồm 2 số nguyên N và Q.
* N dòng tiếp theo, mỗi dòng cho biết tọa độ x[i], y[i] của trạm thứ i.
* Q dòng tiếp theo, mỗi dòng thuộc một trong hai dạng dưới đây:
  + Truy vấn “U I X Y” làm thay đổi tọa độ điểm thứ I thành tọa độ mới (X, Y)
  + Truy vấn “Q I J” cho biết một thí sinh của anh Bell phải chạy từ trạm I tới trạm J. Bạn hãy tính thời gian ngắn nhất để thí sinh này hoàn thành cuộc thi.

**Output**

* Với mỗi truy vấn loại 2, hãy in ra đáp án trên một dòng.

**Giới hạn:**

* 1 ≤ N, Q ≤ 100 000.
* -1000 ≤ x[i], y[i] ≤ 1000.
* 30% test có N, Q ≤ 1000.

**Test ví dụ:**

|  |  |
| --- | --- |
| Input | Output |
| 5 5  -4 4  -5 -3  -1 5  -3 4  0 5  Q 1 5  U 4 0 1  U 4 -1 1  Q 2 4  Q 1 4 | 11  8  8 |

**Giải thích test:**

Truy vấn 1: Bỏ qua trạm thứ 2, hành trình 1🡪3🡪4🡪5.

BÀI 12. ĐẾM SỐ TAM GIÁC

Cho mặt phẳng Oxy, có Q truy vấn:

Loại 1 có dạng (1, x, y): thêm một điểm tại tọa độ (x, y). Các điểm thêm vào là phân biệt.

Loại 2 có dạng (2, a, b, u, v): yêu cầu số tam giác được tạo ra từ 3 điểm nằm trong hình chữ nhật có điểm trái dưới là (a, b), điểm phải trên là (u, v) (a ≤ u, b ≤ v).

**Input**

* Dòng đầu tiên là số nguyên Q (1 ≤ Q ≤ 100 000).
* Q dòng tiếp theo, mỗi dòng gồm một trong hai loại truy vấn như trên.
* Các tọa độ nằm trong khoảng từ 1 tới 1000.

**Output**

* Với mỗi truy vấn loại 2, hãy in ra đáp án trên một dòng.

**Test ví dụ:**

|  |  |
| --- | --- |
| Input | Output |
| 8  1 2 2  1 3 5  1 4 2  1 4 5  1 5 4  2 1 1 6 6  1 3 3  2 1 1 6 6 | 10  20 |

BÀI 13. SỐ CẶP NGHỊCH THẾ TRONG MA TRẬN

Cho ma trận A có kích thước N x N. Nhiệm vụ của bạn là đếm số lượng cặp nghịch thế thỏa mãn: x1 ≤ x2, y1 ≤ y2 và A[x2][y2] < A[x1][y1].

**Input**

* Dòng đầu tiên là số lượng bộ test T (T ≤ 10).
* Mỗi test bắt đầu bởi số nguyên N (1 ≤ N ≤ 500).
* N dòng tiếp theo, mỗi dòng gồm N số nguyên mô tả ma trận A (1 ≤ A[i][j] ≤ 10^9).

**Output**

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

**Test ví dụ:**

|  |  |
| --- | --- |
| Input | Output |
| 2  2  7 5  3 1  4  4 7 2 9  6 4 1 7  5 3 8 1  3 2 5 6 | 5  43 |

**PHẦN 4. PERSISTENT TREE**

BÀI 14. SỐ NHỎ THỨ K

Cho dãy số A[] có N phần tử. Có Q truy vấn dạng (u, v, K) yêu cầu bạn tìm số nhỏ thứ K trong dãy con A[u], A[u+1], ..., A[v].

**Input**

* Dòng đầu tiên chứa 2 số N và Q (1 ≤ N ≤ 100 000, 1 ≤ Q ≤ 5000).
* Dòng tiếp chứa N số nguyên A[i] (-10^9 ≤ A[i] ≤ 10^9), các số là phân biệt.
* Q dòng tiếp, mỗi dòng gồm 3 số nguyên u, v, K (1 ≤ K ≤ v-u+1).

**Output**

* Với truy vấn, in ra đáp án tìm được trên một dòng.

**Test ví dụ:**

|  |  |
| --- | --- |
| Input | Output |
| 7 3  1 5 2 6 3 7 4  2 5 3  4 4 1  1 7 3 | 5  6  3 |

BÀI 15. TRUY VẤN TRÊN CÂY

Cho một cây có N đỉnh. Mỗi đỉnh có trọng số w[i]. Có M truy vấn, mỗi truy vấn có dạng (u, v, k) yêu cầu bạn tìm đỉnh có trọng số nhỏ thứ k trên đường đi từ u tới v.

**Input**

* Dòng đầu tiên chứa 2 số N và M (1 ≤ N, M ≤ 100 000).
* Dòng tiếp gồm N số nguyên w[i] (1 ≤ w[i] ≤ 10^9)
* N-1 dòng tiếp, mỗi dòng gồm 2 số nguyên u và v cho biết đỉnh u nối với đỉnh v.
* M câu truy vấn, mỗi dòng có dạng u v k (k đảm bảo nằm trong phạm vi độ dài đường đi từ u tới v).

**Output**

* Với truy vấn, in ra đáp án tìm được trên một dòng.

**Test ví dụ:**

|  |  |
| --- | --- |
| Input | Output |
| 8 5  105 2 9 3 8 5 7 7  1 2  1 3  1 4  3 5  3 6  3 7  4 8  2 5 1  2 5 2  2 5 3  2 5 4  7 8 2 | 2  8  9  105  7 |