**ĐỀ VÀ ĐÁP ÁN THỰC HÀNH LẦN 2**

**BÀI 1. DSA\_P052. ĐÁNH SỐ THỨ TỰ CẶP DẤU NGOẶC**

Cho một biểu thức đúng, tức là các dấu ngoặc đơn đều đầy đủ mở và đóng, đảm bảo đúng thứ tự. Hãy viết chương trình đánh số các cặp dấu ngoặc theo thứ tự xuất hiện, tính từ 1.

Ví dụ với biểu thức                                              (a + (b \*c) ) + (d/e)

ta có thứ tự của các cặp ‘(‘, ‘)’ là                         1  2  2  1  3  3

**Input**:

* Dòng đầu tiên đưa vào số lượng bộ test T (không quá 100).
* Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test là một biểu thức số học được đưa vào trên một dòng, độ dài không quá 106.

**Output**:

* Đưa ra kết quả mỗi test theo từng dòng.

**Ví dụ:**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 2  (a + (b \*c) ) + (d/e)  ( ( () ) ( () ) ) | 1  2  2  1  3  3  1 2  3  3  2  4  5  5  4  1 |

**Đáp án**

<https://github.com/NguyenHoangHaiD21PTIT/INT1306_CAU_TRUC_DU_LIEU_VA_GIAI_THUAT_HK20222/blob/main/B%C3%80I%20KI%E1%BB%82M%20TRA%20S%E1%BB%90%202%20(75%20PH%C3%9AT%20-%2010%20B%C3%80I)%20(Ng%C4%83n%20x%E1%BA%BFp%2C%20H%C3%A0ng%20%C4%91%E1%BB%A3i%2C%20Duy%E1%BB%87t%20%C4%91%E1%BB%93%20th%E1%BB%8B)/DE%20THI%20NGAY%2012.%2004/DSA_P052.%20DANH%20SO%20THU%20TU%20CAP%20DAU%20NGOAC.cpp>

**BÀI 2. DSA\_P053. ĐOẠN LIÊN TIẾP**

Cho dãy số A[] có N phần tử. Với mỗi vị trí thứ i trong dãy, hãy tính độ dài của đoạn liên tiếp tính từ i trở về phía trước mà các giá trị đều nhỏ hơn hoặc bằng A[i].

**Input:**Dòng đầu ghi số bộ test (không quá 10). Mỗi test có 2 dòng.

* Dòng đầu tiên gồm 1 số nguyên N (1 ≤ N ≤ 105).
* Dòng tiếp theo gồm N số nguyên A1, A2, …, AN (1 ≤ A[i] ≤ 106).

**Output**

* Với mỗi bộ test, in ra dãy kết quả trên một dòng.

**Ví dụ:**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 1  7  100 80 60 70 60 75 85 | 1 1 1 2 1 4 6 |

**Đáp án**

<https://github.com/NguyenHoangHaiD21PTIT/INT1306_CAU_TRUC_DU_LIEU_VA_GIAI_THUAT_HK20222/blob/main/B%C3%80I%20TH%E1%BB%B0C%20H%C3%80NH%20S%E1%BB%90%202%20(Ch%E1%BB%A7%20%C4%91%E1%BB%81%20Ng%C4%83n%20x%E1%BA%BFp%2C%20H%C3%A0ng%20%C4%91%E1%BB%A3i%2C%20Duy%E1%BB%87t%20%C4%91%E1%BB%93%20th%E1%BB%8B%20v%C3%A0%20C%C3%A2y%20nh%E1%BB%8B%20ph%C3%A2n)%20(Thi%20ng%C3%A0y%2007-05-2023)/DSA_P053.%20DOAN%20LIEN%20TIEP.cpp>

**BÀI 3. DSA\_P057. SỐ LỘC PHÁT ĐỐI XỨNG CHẴN**

Trong quan điểm người Việt, một số nguyên dương được gọi là “lộc phát” nếu chỉ có hai chữ số là 6 và/hoặc 8. Một số lộc phát X được gọi là đối xứng chẵn nếu:

* Số chữ số của X là chẵn
* X thỏa mãn tính chất đối xứng.

Hãy liệt kê N số lộc phát đối xứng chẵn đầu tiên theo thứ tự từ nhỏ đến lớn.

**Input**

* Dòng đầu tiên đưa vào số lượng test T.
* Dòng tiếp theo đưa vào các bộ test. Mỗi bộ test được là một số N được viết trên một dòng.
* T, N thỏa mãn ràng buộc: 1 ≤ T ≤ 100; 1 ≤ N ≤ 104.

**Output**

Đưa ra kết quả mỗi test theo từng dòng. Các số cách nhau đúng một khoảng trống.

**Ví dụ**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 2  4  10 | 66 88 6666 6886  66 88 6666 6886 8668 8888 666666 668866 686686 688886 |

**Đáp án**

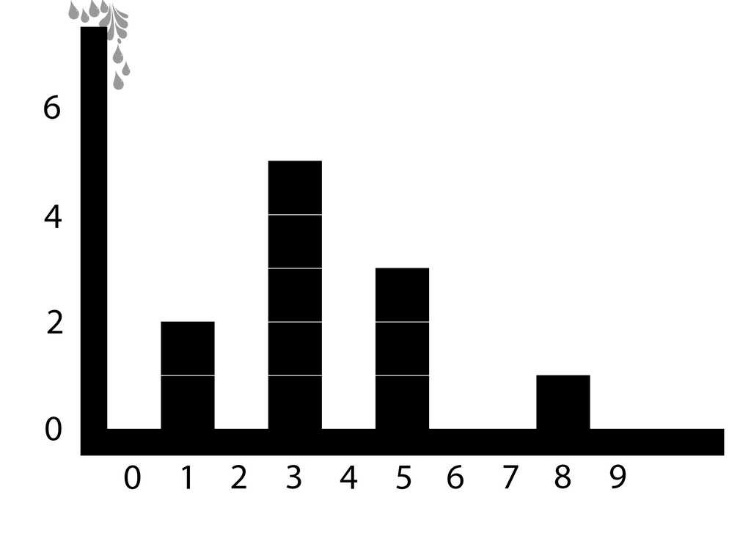
<https://github.com/NguyenHoangHaiD21PTIT/INT1306_CAU_TRUC_DU_LIEU_VA_GIAI_THUAT_HK20222/blob/main/B%C3%80I%20KI%E1%BB%82M%20TRA%20S%E1%BB%90%202%20(75%20PH%C3%9AT%20-%2010%20B%C3%80I)%20(Ng%C4%83n%20x%E1%BA%BFp%2C%20H%C3%A0ng%20%C4%91%E1%BB%A3i%2C%20Duy%E1%BB%87t%20%C4%91%E1%BB%93%20th%E1%BB%8B)/DE%20THI%20NGAY%2012.%2004/DSA_P057.%20S%E1%BB%90%20L%E1%BB%98C%20PH%C3%81T%20%C4%90%E1%BB%90I%20X%E1%BB%A8NG%20CH%E1%BA%B4N.cpp>

**BÀI 4. DSA\_P064. XÂY ĐẬP**

Trên thượng lưu sông High River, một con đập lớn đang được xây dựng. Do dòng sông chảy xiết, các kĩ sư thiết kế N bức tường chắn nước tại các vị trí L[i], có độ cao tương ứng bằng H[i]. Thiết diện mặt cắt của con đập được biểu diễn như hình vẽ. Mỗi bức tường có độ dày bằng 1 đơn vị.

Do tính phức tạp của con đập, một số bức tường vẫn đang được xây dựng tiếp theo ở phía dưới. Do đó, các kĩ sư cần có hệ thống cảnh báo để xác định giới hạn vùng an toàn cho công việc xây dựng ở phía sau.

Giả sử lưu lượng nước đổ về là K đơn vị thể tích. Các bạn giúp các kĩ sư hãy tính toán xem vị trí bức tường cuối cùng sẽ bị nước tràn qua?



**Input**

* Dòng đầu tiên là số lượng bộ test T (T ≤ 20).
* Mỗi test bắt đầu bởi số lượng bức tường N (N ≤ 105).
* Dòng thứ hai gồm N số nguyên L[] mô tả vị trí của các bức tường (1 ≤ L[i] ≤ 109, L[i] > L[i-1]+1).
* Dòng thứ ba gồm N số nguyên H[] mô tả chiều cao của các bức tường (1 ≤ H[i] ≤ 105).
* Tiếp theo là số lượng truy vấn Q (Q ≤ 105).
* Q dòng tiếp theo, mỗi dòng gồm một số nguyên K (1 ≤ K ≤ 1015).

**Output**

* Với mỗi truy vấn, hãy in ra đáp án trên một dòng. Nếu bức tường thứ nhất không bị vượt qua, in ra 0.

**Ví dụ:**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 1  4  1 3 5 8  2 5 3 1  3  17  3  13 | 3  1  1 |

Giải thích test:

|  |  |  |
| --- | --- | --- |
| 在这里插入图片描述 | 在这里插入图片描述 | 在这里插入图片描述 |

**Đáp án**

<https://github.com/NguyenHoangHaiD21PTIT/INT1306_CAU_TRUC_DU_LIEU_VA_GIAI_THUAT_HK20222/blob/main/KI%E1%BB%82M%20TRA%2C%20KH%E1%BA%A2O%20S%C3%81T%20D22/DSA_P064.%20XAY%20DAP.cpp>

**BÀI 5. DSA\_P065. LOẠI BỎ DẤU NGOẶC**

Cho một biểu thức đúng và thỏa mãn:

- Các biến trong biểu thức chỉ chứa các chữ cái viết hoa.

- Các toán tử trong biểu thức là ‘+’ hoặc ‘-’

Hãy loại bỏ các dấu ngoặc thừa mà vẫn giữ nguyên ý nghĩa của biểu thức.

**Input:**

- Dòng đầu tiên chứa số biểu thức M (1<=M<=10).

- M dòng tiếp theo, mỗi dòng là một biểu thức đúng, có thể có các dấu cách tùy ý trong mỗi dòng. Độ dài mỗi dòng (bao gồm cả dấu cách) không quá 255 kí tự.

**Output:**

Với mỗi biểu thức, in ra trên một dòng biểu thức không có các dấu ngoặc thừa.

Chú ý: Thứ tự của các toán hạng trong biểu thức kết quả và biểu thức đầu vào phải giống nhau. Các dấu cách thừa nếu có cũng phải được loại bỏ.

**Ví dụ:**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 3  (A  -  B + C)-(A+   (B–C))  -  (C-   (D- E) )  ((A)-  ((B  )    ))  A-(   B+C) | A-B+C-(A+B-C)-(C-(D-E))  A-B  A-(B+C) |

**Đáp án:**

<https://github.com/NguyenHoangHaiD21PTIT/INT1306_CAU_TRUC_DU_LIEU_VA_GIAI_THUAT_HK20222/blob/main/B%C3%80I%20TH%E1%BB%B0C%20H%C3%80NH%20S%E1%BB%90%202%20(Ch%E1%BB%A7%20%C4%91%E1%BB%81%20Ng%C4%83n%20x%E1%BA%BFp%2C%20H%C3%A0ng%20%C4%91%E1%BB%A3i%2C%20Duy%E1%BB%87t%20%C4%91%E1%BB%93%20th%E1%BB%8B%20v%C3%A0%20C%C3%A2y%20nh%E1%BB%8B%20ph%C3%A2n)%20(Thi%20ng%C3%A0y%2007-05-2023)/DSA_P065.%20LOAI%20BO%20DAU%20NGOAC.cpp>

**BÀI 6. DSA\_P060. CHỮ SỐ NGUYÊN TỐ**

Chúng ta đều biết chỉ có 4 chữ số nguyên tố là 2, 3, 5, 7. Hãy liệt kê tất cả các số có ít nhất 4 chữ số nhưng không quá N chữ số và thỏa mãn tất cả các điều kiện sau:

* Chỉ có các chữ số 2, 3, 5, 7
* Có đầy đủ 4 chữ số 2, 3, 5, 7
* Không phải là số chẵn.

**Input**. Chỉ có 1 dòng ghi số N (3 < N < 10)

**Output**. Ghi ra lần lượt các số thỏa mãn theo thứ tự tăng dần, mỗi số trên một dòng

**Ví dụ**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 4 | 2357  2375  2537  2573  2735  2753  3257  3275  3527  3725  5237  5273  5327  5723  7235  7253  7325  7523 |

**Đáp án**

<https://github.com/NguyenHoangHaiD21PTIT/INT1306_CAU_TRUC_DU_LIEU_VA_GIAI_THUAT_HK20222/blob/main/B%C3%80I%20KI%E1%BB%82M%20TRA%20S%E1%BB%90%201%20(60%20PH%C3%9AT%20-%204%20B%C3%80I)%20(Sinh%20k%E1%BA%BF%20ti%E1%BA%BFp%2C%20Quay%20lui%20-%20Nh%C3%A1nh%20c%E1%BA%ADn%2C%20Gi%E1%BA%A3i%20thu%E1%BA%ADt%20tham%20lam%2C%20Chia%20v%C3%A0%20tr%E1%BB%8B)/DSA_P060.%20CHU%20SO%20NGUYEN%20TO.cpp>

**BÀI 7. DSA\_P058. SỐ BƯỚC DI CHUYỂN ÍT NHẤT**

Cho ma trận A kích thước N\*M.

Hãy tìm **số bước đi ít nhất** để di chuyển từ vị trí A[1][1] đến vị trí A[N][M].

Biết rằng mỗi bước từ vị trí (i, j) ta có thể di chuyển theo một trong ba hướng:

* Hướng xuống dưới với số ô di chuyển là hiệu hai giá trị A[i][j] và A[i+1][j]
* Hướng sang phải với số ô di chuyển là hiệu hai giá trị A[i][j] và A[i][j+1]
* Hướng chéo xuống với số ô di chuyển là hiệu của hai giá trị A[i][j] và A[i+1][j+1]

**Input:**

* Dòng đầu tiên đưa vào số lượng test T.
* Dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm hai phần: phần thứ nhất là hai số N, M; phần thứ hai là các phần tử của ma trận A[][]; các số được viết cách nhau một vài khoảng trống.
* T, N, M, A[i][j] thỏa mãn ràng buộc: 1≤T≤100; 1≤ N, M, A[i][j]≤103.

**Output:**

* Đưa ra kết quả mỗi test theo từng dòng.
* Nếu không tìm được đường đi ghi ra -1

**Ví dụ:**

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

**Đáp án:**

<https://github.com/NguyenHoangHaiD21PTIT/INT1306_CAU_TRUC_DU_LIEU_VA_GIAI_THUAT_HK20222/blob/main/B%C3%80I%20TH%E1%BB%B0C%20H%C3%80NH%20S%E1%BB%90%202%20(Ch%E1%BB%A7%20%C4%91%E1%BB%81%20Ng%C4%83n%20x%E1%BA%BFp%2C%20H%C3%A0ng%20%C4%91%E1%BB%A3i%2C%20Duy%E1%BB%87t%20%C4%91%E1%BB%93%20th%E1%BB%8B%20v%C3%A0%20C%C3%A2y%20nh%E1%BB%8B%20ph%C3%A2n)%20(Thi%20ng%C3%A0y%2007-05-2023)/DSA_P058.%20SO%20BUOC%20DI%20CHUYEN%20IT%20NHAT.cpp>

**BÀI 8. DSA\_P069. VUA VÀ QUÂN MÃ**

Vua Titan và các hiệp sĩ bàn tròn thường gặp nhau vào đầu năm mới để ăn mừng tình bạn của họ. Để tưởng nhớ sự kiện này, chúng ta xem xét một trò chơi, trong đó có một quân vua và **một** quân mã được đặt ngẫu nhiên trên các ô riêng biệt. Bàn cờ có kích thước N×N, trên bàn cờ có một số ô cấm những ô còn lại là những ô tự do – ô có thể di chuyển vào được. Các ô đặt quân mã và quân vua đang đứng ở các ô tự do.

|  |  |
| --- | --- |
| chess king moves for Sale OFF 72% | chess board knight moves for Sale OFF 67% |

Tại mỗi bước tất cả các quân đều phải di chuyển theo quy tắc và không được đi vào ô cấm, hãy tìm cách di chuyển để chúng gặp nhau nhanh nhất.

**Input:**

Dòng đầu tiên là số nguyên N (N ≤ 100).

N dòng tiếp theo, mỗi dòng gồm 1 xâu có N ký tự, mô tả bàn cờ. “.” Thể hiện ô trống, “#” thể hiện ô cấm, “T” thể hiện vị trí vua đang đứng và “M” thể hiện vị trí quân mã đang đứng.

**Output:**

In ra đáp án là số bước ít nhất để các quân gặp nhau. Nếu không thể gặp nhau, in ra -1.

**Ví dụ:**

|  |  |
| --- | --- |
| Input | Output |
| 5  M....  .....  .#...  .#..#  ...#T | 2 |

**Đáp án**

<https://github.com/NguyenHoangHaiD21PTIT/INT1306_CAU_TRUC_DU_LIEU_VA_GIAI_THUAT_HK20222/blob/main/DE%20LUYEN%20TAP%20TUAN%201/CODE%20THAM%20KH%E1%BA%A2O/CODE%20CONTEST%20TH%E1%BB%A8%207%20C%C3%81C%20TU%E1%BA%A6N%202%2C%203%2C%204/DSA_P069.%20VUA%20VA%20QUAN%20MA.cpp>

**BÀI 9. DSA\_P085. BIẾN ĐỔI DÃY SỐ**

Cho dãy số nguyên A[] có N phần tử. Một dãy số được coi là đẹp nếu trong dãy không có bất cứ cặp số cạnh nhau nào bằng nhau (tức là A[i] # A[i+1] với mọi chỉ số i từ 0 đến N-2).

Giả sử chi phí để thay đổi giá trị A[i] tăng 1 đơn vị là B[i].

Hãy tính chi phí nhỏ nhất để biến đổi dãy số A[] ban đầu về dạng thỏa mãn tính chất “đẹp”.

**Bạn chỉ được phép tăng A[i] lên 0, 1 hoặc 2 đơn vị**

**Input**

Dòng đầu ghi số bộ test T (1 < T ≤ 3\*105).

Mỗi bộ test bắt đầu với số nguyên dương N (1 < N ≤ 3\*105).

Tiếp theo là N dòng, mỗi dòng ghi 2 số A[i] và B[i]. Các giá trị đều nguyên dương và không quá 9 chữ số.

Dữ liệu vào đảm bảo kết quả tính được của mỗi test đều không quá 18 chữ số.

**Output**

Với mỗi bộ test, ghi ra chi phí nhỏ nhất có thể trên một dòng.

**Ví dụ**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 3  3  2 4  2 1  3 5  3  2 3  2 10  2 6  4  1 7  3 3  2 6  1000000000 2 | 2  9  0 |

**Đáp án**

<https://github.com/NguyenHoangHaiD21PTIT/INT1306_CAU_TRUC_DU_LIEU_VA_GIAI_THUAT_HK20222/blob/main/DE%20LUYEN%20TAP%20TUAN%201/CODE%20THAM%20KH%E1%BA%A2O/CODE%20CONTEST%20TH%E1%BB%A8%207%20C%C3%81C%20TU%E1%BA%A6N%202%2C%203%2C%204/DSA_P085.%20BIEN%20DOI%20DAY%20SO.cpp>

**BÀI 10. DSA\_P110. TỔNG GIÁ TRỊ LỚN NHẤT**

Một tên trộm đột nhập vào một cửa hiệu tìm thấy n món hàng có trọng lượng và giá trị khác nhau, nhưng hắn chỉ mang theo một cái túi có sức chứa về trọng lượng tối đa là M. Vậy tên trộm nên bỏ vào túi những món nào để đạt giá trị cao nhất trong khả năng mà hắn có thể mang đi được.

**Input**

* Dòng đầu tiên chứa hai số nguyên dương n và M (1 ≤ n, M ≤ 5000).
* n dòng tiếp theo, mỗi dòng chứa hai số nguyên dương x và y mô tả một đồ vật có trọng lượng x và giá trị y (1 ≤ x ≤ M, 1 ≤ y ≤ 10000).

**Output**

* In ra tổng giá trị lớn nhất đạt được.

**Ví dụ**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 10 50  33 6  19 3  12 8  22 7  18 3  34 10  14 10  21 9  26 10  40 4 | 27 |

**Đáp án**

<https://github.com/NguyenHoangHaiD21PTIT/INT1306_CAU_TRUC_DU_LIEU_VA_GIAI_THUAT_HK20222/blob/main/BAI%20THUC%20HANH%20SO%201%20(Ch%E1%BB%A7%20%C4%91%E1%BB%81%20Sinh%20k%E1%BA%BF%20ti%E1%BA%BFp%2C%20Quay%20lui%20-%20Nh%C3%A1nh%20c%E1%BA%ADn%2C%20Gi%E1%BA%A3i%20thu%E1%BA%ADt%20tham%20lam%2C%20Chia%20v%C3%A0%20tr%E1%BB%8B%2C%20Quy%20ho%E1%BA%A1ch%20%C4%91%E1%BB%99ng)/CONTEST%20DP%20(12%20-%2003%20-%202023)/DSA_P110.%20TONG%20GIA%20TRI%20LON%20NHAT.cpp>

**BÀI 11. DSA\_P207. TÍNH TOÁN GIÁ TRỊ BIỂU THỨC**

Biểu thức dưới dạng hậu tố là phép biểu diễn biểu thức trong đó phép toán được đặt sau hai toán hạng.

Ví dụ AB+CD-\* (tương ứng với biểu thức trung tố (A+B)\*(C-D).

Biểu thức dưới dạng tiền tố là phép biểu diễn biểu thức trong đó phép toán được đặt trước hai toán hạng.

Ví dụ + \* A B / C D tương ứng với biểu thức trung tố A \* B + C / D

Cho một biểu thức ở một trong hai dạng hậu tố hoặc tiền tố. Hãy tính giá trị biểu thức đó.

**Input:**

Dòng đầu tiên là số lượng bộ test T (T ≤ 50).

Mỗi test bắt đầu bởi số nguyên N (N ≤ 1000), là số lượng toán tử và toán hạng.

Dòng tiếp theo là biểu thức hậu tố hoặc tiền tố với N toán tử và toán hạng. Các toán hạng và toán tử cách nhau bởi một dấu cách.

Các toán hạng có thể có nhiều chữ số, có thể là số âm. Các toán tử chỉ bao gồm các phép toán cộng, trừ, nhân, chia, trong đó phép chia thực hiện trên số nguyên.

Input đảm bảo kết quả phép toán có giá trị tuyệt đối không vượt quá 1018.

**Output:**

Với mỗi test, in ra giá trị biểu thức tìm được.

**Ví dụ:**

|  |  |
| --- | --- |
| **Input:** | **Output** |
| 3  7  2 3 1 \* + 9 –  7  - + 8 \* 7 5 9  3  -10 -4 + | -4  34  -14 |

**Đáp án**

<https://github.com/NguyenHoangHaiD21PTIT/INT1306_CAU_TRUC_DU_LIEU_VA_GIAI_THUAT_HK20222/blob/main/THI%20THU%20CUOI%20KY/KHAO%20SAT%20LAN%201%20-%2019.05.2023/DSA_P207.%20TINH%20TOAN%20GIA%20TRI%20BIEU%20THUC.cpp>

**BÀI 12. DSA\_P070. THÍ NGHIỆM VẬT LÝ**

Có một chiếc đèn laze tại vị trí (xA, yA) và bạn cần phải chiếu sáng vị trí (xB, yB). Trên hệ thống mô phỏng có N vị trí đã gắn sẵn giá gương, bạn được phép lựa chọn có đặt gương vào các vị trí này hay không. Nếu không có gương, tia laze sẽ tiếp tục truyền thẳng theo hướng song song với trục Ox hoặc trục Oy. Nếu bạn đặt gương, ánh sáng laze sẽ bị bẻ vuông 90 độ (sang trái, hoặc phải).

Nhiệm vụ của bạn là hãy xác định số lượng gương ít nhất cần sử dụng, sao cho ánh sáng laze có thể chiếu sáng tới được vị trí B.

**Input**

Dòng đầu tiên là số nguyên N (1 ≤ N ≤ 100 000) và xA, yA, xB, yB.

N dòng tiếp theo, mỗi dòng gồm tọa độ của các vị trí đặt giá gương. (Vị trí đặt gương có thể trùng với A hoặc B, và không có vị trí nào đặt được 2 gương).

Các tọa độ nằm trong phạm vi từ 0 đến 109.

**Output**

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

Nếu vị trí B không thể được chiếu sáng, in ra -1.

**Ví dụ:**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 4 0 0 9 3  1 7  0 3  3 0  3 3 | 1 |

Giải thích test: Đặt gương tại vị trí (0, 3).

**Đáp án**

<https://github.com/NguyenHoangHaiD21PTIT/INT1306_CAU_TRUC_DU_LIEU_VA_GIAI_THUAT_HK20222/blob/main/KI%E1%BB%82M%20TRA%2C%20KH%E1%BA%A2O%20S%C3%81T%20D22/DSA_P070.%20THI%20NGHIEM%20VAT%20LY.cpp>