

TỔNG QUAN ĐỀ THI

	Tiêu đề	File nộp	File dữ liệu	File kết quả	Thời gian	Bộ nhớ
Bài 1	Dãy đèn	LIGHT.*	LIGHT.INP	LIGHT.OUT	1 giây	1024 MB
Bài 2	Quà Noel	GIFT.*	GIFT.INP	GIFT.OUT	1 giây	1024 MB
Bài 3	Bài đăng	POST.*	POST.INP	POST.OUT	1 giây	1024 MB

- Dấu * được thay thế bởi PAS hoặc CPP hoặc PY tương ứng với ngôn ngữ lập trình Pascal hoặc C++ hoặc Python.
- Mỗi bài bao gồm nhiều subtask, mỗi subtask bao gồm nhiều test đơn, điểm của thí sinh được tính theo từng test đơn.
- Thí sinh không được phép sử dụng các định hướng biên dịch chương trình có các từ khoá sau: pragma, optimize, target, O3, Ofast, unroll-loops, avx, avx2, fma, omit-frame-pointer.
- Bài của thí sinh sẽ được chấm điểm trên máy chấm có CPU tương đương với Intel Core i9-13900K 3.20 GHz. Thí sinh sử dụng thông tin máy chấm để tham khảo và đánh giá thời gian chạy đáp ứng yêu cầu đề bài.

Hãy lập trình giải các bài toán sau:

Bài 1. Dây đèn (7,0 điểm)

Giáng sinh năm nay, khu phố VOI được trang trí bởi một dàn N bóng đèn, các bóng đèn được đánh số từ 1 đến N . Ban đầu, mỗi bóng đèn ở một trong 3 trạng thái *tắt*, *vàng* hoặc *đỏ* và được điều khiển bởi một công tắc vòng tròn hoạt động như sau:

- Khi xoay theo chiều kim đồng hồ một nấc, trạng thái bóng đèn đó thay đổi theo qui tắc:
 - Nếu đang ở trạng thái *tắt* thì chuyển sang trạng thái *vàng*;
 - Nếu đang ở trạng thái *vàng* thì chuyển sang trạng thái *đỏ*;
 - Nếu đang ở trạng thái *đỏ* thì chuyển sang trạng thái *tắt*.
- Khi xoay ngược chiều kim đồng hồ một nấc, trạng thái bóng đèn đó thay đổi theo qui tắc:
 - Nếu đang ở trạng thái *tắt* thì chuyển sang trạng thái *đỏ*;
 - Nếu đang ở trạng thái *đỏ* thì chuyển sang trạng thái *vàng*;
 - Nếu đang ở trạng thái *vàng* thì chuyển sang trạng thái *tắt*.

Ban quản trị VOI nhận được Q yêu cầu khảo sát, mỗi yêu cầu gồm hai giá trị L và R với $1 \leq L \leq R \leq N$. Cụ thể, mỗi yêu cầu khảo sát cần tìm một dãy các thao tác để đưa tất cả các bóng đèn từ L đến R về trạng thái *tắt*, mỗi thao tác thay đổi các bóng đèn như sau:

Chọn $X + Y$ bóng đèn phân biệt trong đoạn từ L đến R , trong đó chọn X bóng đèn để xoay công tắc theo chiều kim đồng hồ một nấc và Y bóng đèn còn lại để xoay công tắc ngược chiều kim đồng hồ một nấc.

Lưu ý: Các yêu cầu khảo sát là độc lập, nghĩa là mỗi yêu cầu đều được xử lý trên trạng thái ban đầu của dãy đèn. Giá trị X và Y là cố định cho tất cả Q yêu cầu. Các công tắc vòng tròn được thiết kế để có thể xoay không giới hạn theo cả hai chiều.

Yêu cầu: Với mỗi yêu cầu khảo sát, hãy tìm số thao tác ít nhất để đưa toàn bộ các bóng đèn trong đoạn từ L đến R của dãy đèn về trạng thái *tắt*.

Dữ liệu

Vào từ file văn bản LIGHT.INP:

- Dòng đầu chứa 4 số nguyên không âm N, Q, X, Y ($0 < X + Y \leq 5$; $X + Y \leq N \leq 666$; $1 \leq Q \leq 10^5$).
- Dòng thứ hai chứa N số nguyên, số thứ i có giá trị 0, 1 hoặc 2 tương ứng với trạng thái ban đầu của bóng đèn thứ i là *tắt*, *vàng* hoặc *đỏ* ($1 \leq i \leq N$).
- Mỗi dòng trong số Q dòng tiếp theo chứa hai số nguyên dương L và R mô tả một yêu cầu. Dữ liệu bảo đảm $R - L + 1 \geq X + Y$ trong tất cả các yêu cầu khảo sát.

Các số trên cùng một dòng cách nhau bởi dấu cách.

Kết quả

Ghi ra file văn bản LIGHT.OUT:

- Gồm Q dòng, mỗi dòng một số nguyên là số lượng thao tác ít nhất tìm được cho yêu cầu khảo sát tương ứng hoặc -1 nếu không tồn tại phương án.

Ví dụ

LIGHT.INP	LIGHT.OUT	Giải thích
5 3 1 1 2 2 2 2 1 1 5 2 4 3 5	3 2 -1	Yêu cầu khảo sát đầu tiên cần ít nhất 3 thao tác. Hình sau minh họa một dãy thao tác tối ưu: 22221 22200 01200 00000  Xoay theo chiều kim đồng hồ  Xoay ngược chiều kim đồng hồ

Chấm điểm

- Subtask 1 (30% số điểm): $X = Y = 1; N \leq 10; Q = 1$.
- Subtask 2 (20% số điểm): $X = Y = 1; Q = 1$.
- Subtask 3 (34% số điểm): $Q \leq 5$.
- Subtask 4 (16% số điểm): Không có ràng buộc nào thêm.

Bài 2. Quà Noel (7,0 điểm)

Ông già Noel đã chuẩn bị N loại quà, loại quà thứ i ($1 \leq i \leq N$) có S_i món và khối lượng mỗi món là W_i gram. Ông dự định đóng gói các túi quà để phát cho trẻ em ở VOI sao cho mỗi túi có đúng K món và không có hai món nào cùng một loại quà. Tuy nhiên xe tuần lộc của ông chỉ có thể mang được không quá M gram quà.

Để chuẩn bị chu đáo trước khi lên đường, ông già Noel đưa ra một loạt Q câu hỏi cho trợ lý của ông tính toán, mỗi câu hỏi thuộc một trong hai loại sau:

- Loại 1: Cho biết 3 giá trị M, K, T , hãy kiểm tra xem có thể tạo ra được T túi quà mà tổng khối lượng các món quà trong cả T túi đó không vượt quá M gram hay không?
- Loại 2: Cho biết 2 giá trị M, K , hỏi có thể tạo ra được nhiều nhất bao nhiêu túi quà mà tổng khối lượng các món quà trong tất cả các túi đó không vượt quá M gram?

Yêu cầu: Hãy giúp trợ lý trả lời từng câu hỏi của ông già Noel.

Dữ liệu

Vào từ file văn bản GIFT.INP:

- Dòng đầu chứa 2 số nguyên N, Q ($1 \leq N, Q \leq 10^5$).
- Dòng thứ hai chứa N số nguyên dương $W_1, W_2, \dots, W_N$ có giá trị không quá 10^5 .
- Dòng thứ ba chứa N số nguyên dương $S_1, S_2, \dots, S_N$ có giá trị không quá 10^5 .
- Mỗi dòng trong số Q dòng tiếp theo thể hiện một câu hỏi thuộc một trong hai loại:
 - Loại 1, gồm bốn số nguyên dương: $1 \ M \ K \ T$ ($1 \leq M, K, T \leq 10^9$).
 - Loại 2, gồm ba số nguyên dương: $2 \ M \ K$ ($1 \leq M, K \leq 10^9$).

Các số trên cùng một dòng cách nhau bởi dấu cách.

Kết quả

Ghi ra file văn bản GIFT.OUT:

- Gồm Q dòng, mỗi dòng thể hiện câu trả lời cho một câu hỏi tương ứng:
 - Loại 1: Ghi ra 1 nếu có thể tạo được T túi quà, trái lại ghi ra 0.
 - Loại 2: Ghi ra một số nguyên là số lượng túi quà nhiều nhất tạo được.

Ví dụ

GIFT.INP	GIFT.OUT	Giải thích
5 3	0	- Với câu hỏi 1: Không có cách nào chọn được hai túi, mỗi túi có 3 món mà tổng khối lượng không quá 19 gram.
8 3 4 10 8	1	- Với câu hỏi 2: Tạo ra được nhiều nhất một túi quà. Một phương án là túi quà gồm một món quà loại 2, một món quà loại 3 và một món quà loại 4. Tổng khối lượng là 17 gram.
5 8 2 9 10	3	- Với câu hỏi 3: Tạo ra được nhiều nhất ba túi quà. Mỗi túi gồm đúng một món quà loại 2.
1 19 3 2		
2 19 3		
2 9 1		

Chấm điểm

- Subtask 1 (25% số điểm): $S_i = 1, \forall i = 1, 2, \dots, N$.
- Subtask 2 (20% số điểm): $N, Q \leq 1000$ và chỉ có câu hỏi loại 1.
- Subtask 3 (15% số điểm): $N, Q \leq 1000$.
- Subtask 4 (20% số điểm): Chỉ có câu hỏi loại 1.
- Subtask 5 (20% số điểm): Không có ràng buộc nào thêm.

Bài 3. Bài đăng (6,0 điểm)

Alice là kỹ sư phát triển nền tảng mạng xã hội VOI giúp cho cộng đồng lập trình viên tương tác với nhau. Có N bài đăng được đánh số từ 1 đến N theo trình tự thời gian đăng bài. Alice phân loại các bài đăng theo chủ đề, bài đăng thứ i ($1 \leq i \leq N$) có chủ đề là A_i .

Alice định nghĩa một đoạn $[L, R]$ gồm các bài đăng liên tiếp từ chỉ số L đến chỉ số R ($1 \leq L \leq R \leq N$) là *toàn vẹn* nếu với mỗi chủ đề t , hoặc không có bài nào có chủ đề t nằm trong đoạn $[L, R]$ hoặc tất cả các bài có chủ đề t đều nằm trong đoạn này.

Yêu cầu: Vào dịp Noel, Alice công bố dữ liệu chủ đề của N bài đăng và mở cuộc thi trên VOI cho các lập trình viên tham gia. Alice lần lượt đưa ra Q truy vấn, mỗi truy vấn cho biết một cặp số nguyên U, V ($1 \leq U \leq V \leq N$) và yêu cầu thí sinh đếm số đoạn $[L, R]$ là *toàn vẹn* với $U \leq L \leq R \leq V$.

Cụ thể, với mỗi truy vấn, cần đếm xem có bao nhiêu cặp L, R thỏa mãn:

- $U \leq L \leq R \leq V$;
- Không tồn tại hai giá trị i, j nào sao cho:
 - * $(1 \leq i < L)$ hoặc $(R < i \leq N)$;
 - * $L \leq j \leq R$;
 - * $A_i = A_j$.

Dữ liệu

Vào từ file văn bản POST.INP;

- Dòng đầu chứa hai số nguyên N và Q ($1 \leq N, Q \leq 3 \times 10^5$).
- Dòng thứ hai chứa N số nguyên dương $A_1, A_2, \dots, A_N$ ($1 \leq A_1, A_2, \dots, A_N \leq 10^9$).
- Mỗi dòng trong số Q dòng tiếp theo chứa hai số nguyên dương U và V .

Các số trên cùng một dòng cách nhau bởi dấu cách.

Kết quả

Ghi ra file văn bản POST.OUT:

- Gồm Q dòng, mỗi dòng một số là đáp án cho truy vấn tương ứng.

Ví dụ

POST.INP	POST.OUT	Giải thích
7 2	3	- Truy vấn 1, có 3 đoạn toàn vẹn: [1, 7], [2, 3], [6, 6] - Truy vấn 2, có 2 đoạn toàn vẹn: [2, 3], [6, 6]
1 2 2 3 1 6 3	2	
1 7		
2 6		

Chấm điểm

- Subtask 1 (15% số điểm): $N, Q \leq 50$.
- Subtask 2 (15% số điểm): $N \leq 500$.
- Subtask 3 (10% số điểm): $N \leq 5\,000$.
- Subtask 4 (20% số điểm): $Q = 1; U = 1; V = N$.
- Subtask 5 (20% số điểm): $Q \leq 30\,000$.
- Subtask 6 (20% số điểm): Không có ràng buộc nào thêm.

----- HẾT -----

* Thí sinh **KHÔNG** được sử dụng tài liệu;

* Giám thị **KHÔNG** giải thích gì thêm.

TỔNG QUAN ĐỀ THI

	Tiêu đề	File nộp	File dữ liệu	File kết quả	Thời gian	Bộ nhớ
Bài 4	Tất niên	TET.*	TET.INP	TET.OUT	1 giây	1024 MB
Bài 5	Cây thông	PINE.*	PINE.INP	PINE.OUT	1 giây	1024 MB
Bài 6	Cắm trại	CAMP.*	CAMP.INP	CAMP.OUT	3 giây	1024 MB

- Dấu * được thay thế bởi PAS hoặc CPP hoặc PY tương ứng với ngôn ngữ lập trình Pascal hoặc C++ hoặc Python.
- Mỗi bài bao gồm nhiều subtask, mỗi subtask bao gồm nhiều test đơn, điểm của thí sinh được tính theo từng test đơn.
- Thí sinh không được phép sử dụng các định hướng biên dịch chương trình có các từ khoá sau: pragma, optimize, target, O3, Ofast, unroll-loops, avx, avx2, fma, omit-frame-pointer.
- Bài của thí sinh sẽ được chấm điểm trên máy chấm có CPU tương đương với Intel Core i9-13900K 3.20 GHz. Thí sinh sử dụng thông tin máy chấm để tham khảo và đánh giá thời gian chạy đáp ứng yêu cầu đề bài.

Hãy lập trình giải các bài toán sau:

Bài 4. Tất niên (7,0 điểm)

Trong tiệc tất niên cuối năm, gia đình Alice chuẩn bị một bàn tiệc lớn để chào đón mọi người. Trên bàn có N đĩa thức ăn được bày biện đẹp mắt thành một hàng, được đánh số từ 1 đến N từ trái sang phải. Đĩa thứ i ($1 \leq i \leq N$) có độ hấp dẫn là một số nguyên dương A_i .

Với một đoạn gồm M đĩa ở vị trí liên tiếp của bàn tiệc có độ hấp dẫn tương ứng là dãy $(B_1, B_2, \dots, B_M)$, Alice định nghĩa *độ đa dạng* của dãy đó là số lượng dãy khác nhau có thể nhận được bằng cách thực hiện không quá một phép đảo ngược trên dãy. Một phép đảo ngược là việc chọn một cặp chỉ số (L, R) với $1 \leq L < R \leq M$ và thực hiện đảo ngược dãy $(B_L, B_{L+1}, \dots, B_{R-1}, B_R)$ thành dãy $(B_R, B_{R-1}, \dots, B_{L+1}, B_L)$, các phần tử còn lại giữ nguyên. Hai dãy $(X_1, X_2, \dots, X_M)$ và $(Y_1, Y_2, \dots, Y_M)$ được xem là khác nhau nếu tồn tại chỉ số i ($1 \leq i \leq M$) sao cho $X_i \neq Y_i$. Ví dụ, dãy $(3, 1, 3)$ có độ đa dạng bằng 3 vì có thể tạo được 3 dãy khác nhau với không quá một phép đảo ngược: $(3, 1, 3)$, $(1, 3, 3)$, $(3, 3, 1)$.

Trong buổi tiệc, có Q vị khách lần lượt tới tham dự. Vị khách thứ j ($1 \leq j \leq Q$) đến và lấy chiếc đĩa thứ p_j để dùng bữa. Sau khi một chiếc đĩa được lấy ra, những chiếc còn lại giữ nguyên vị trí ban đầu và tạo thành các đoạn con ngăn cách bởi các vị trí trống do các đĩa

đã được lấy đi. Alice tính độ đa dạng của mỗi đoạn con và muốn biết độ đa dạng lớn nhất trong số chúng.

Yêu cầu: Sau khi mỗi vị khách nhắc một chiếc đĩa đi, hãy tính độ đa dạng lớn nhất trong các đoạn con còn lại.

Dữ liệu

Vào từ file văn bản TET.INP:

- Dòng đầu chứa hai số nguyên N và Q ($1 \leq Q < N \leq 2 \times 10^5$).
- Dòng thứ hai chứa N số nguyên dương $A_1, A_2, \dots, A_N$ ($1 \leq A_i \leq 10^9$ với mọi $i = 1, 2, \dots, N$).
- Dòng thứ j trong số Q dòng tiếp theo chứa một số nguyên dương p_j ($1 \leq p_j \leq N$). Dữ liệu bảo đảm các giá trị p_j là khác nhau.

Các số trên cùng một dòng cách nhau bởi dấu cách.

Kết quả

Ghi ra file văn bản TET.OUT:

- Gồm Q dòng, mỗi dòng một số nguyên duy nhất là kết quả tìm được sau khi vị khách tương ứng nhắc một chiếc đĩa đi.

Ví dụ

TET.INP	TET.OUT	Giải thích
7 3	9	- Sau khi lấy đĩa thứ 2, thu được hai đoạn con là (4) và (3,1,3,2,1), với độ đa dạng lần lượt là 1 và 9. - Sau khi lấy đĩa thứ 5, thu được ba đoạn con là (4), (3,1) và (2,1), với độ đa dạng lần lượt là 1, 2 và 2. - Sau khi lấy đĩa thứ 6, thu được ba đoạn con là (4), (3,1) và (1), với độ đa dạng lần lượt là 1, 2 và 1.
4 1 3 1 3 2 1	2	
2	2	
5		
6		

Chấm điểm

- Subtask 1 (30% số điểm): $Q = 1$; $N \leq 200$.
- Subtask 2 (30% số điểm): $Q = 1$; $N \leq 2000$.
- Subtask 3 (20% số điểm): $Q = 1$.
- Subtask 4 (20% số điểm): Không có ràng buộc nào thêm.

Bài 5. Cây thông (7,0 điểm)

Ông già Noel có một cây thông khổng lồ để trang hoàng cho đêm Giáng sinh. Trên các cành của cây thông, ông treo N tấm thiệp trang trí được đánh số từ 1 đến N . Tấm thiệp i ($1 \leq i \leq N$) có độ lấp lánh được biểu thị bằng số nguyên không âm A_i . Các tấm thiệp được kết nối với nhau bằng $N - 1$ đoạn dây. Đoạn dây thứ i kết nối hai tấm thiệp u_i và v_i ($1 \leq u_i, v_i \leq N$). Hệ thống dây đảm bảo rằng giữa hai tấm thiệp bất kỳ trên cây luôn tồn tại duy nhất một đường kết nối trực tiếp hoặc gián tiếp thông qua các đoạn dây. Nói cách khác, các tấm thiệp và các đoạn dây tạo thành một đồ thị dạng cây có N đỉnh và $N - 1$ cạnh.

Mỗi thời điểm trong đêm Giáng sinh, ông già Noel sẽ thực hiện một phép thuật. Có Q thời điểm lần lượt xảy ra được đánh số từ 1 đến Q . Tại thời điểm t ($1 \leq t \leq Q$), ông thực hiện một phép thuật được mô tả bằng ba số nguyên dương x_t, y_t, w_t với ý nghĩa: Gán $A_v = A_v \% w_t$ với mọi tấm thiệp v nằm trên đường kết nối từ tấm thiệp x_t đến tấm thiệp y_t (bao gồm cả x_t và y_t). Sau khi thực hiện phép thuật, ông định nghĩa *độ đẹp của tấm thiệp v tại thời điểm t* là $(A_v \% t)$; *độ đẹp của cây tại thời điểm t* là tổng độ đẹp của tất cả các tấm thiệp trên cây tại thời điểm t , tức là $(A_1 \% t) + (A_2 \% t) + \dots + (A_N \% t)$. Nhắc lại, $\%$ là phép toán chia lấy dư.

Yêu cầu: Hãy giúp ông già Noel tính độ đẹp của cây sau mỗi lần thực hiện phép thuật.

Dữ liệu

Vào từ file văn bản PINE.INP:

- Dòng đầu chứa hai số nguyên dương N và Q ($1 \leq N, Q \leq 2 \times 10^5$).
- Dòng thứ hai chứa N số nguyên $A_1, A_2, \dots, A_N$ ($0 \leq A_i \leq 2 \times 10^5$).
- Dòng thứ i trong số $N - 1$ dòng tiếp theo chứa hai số nguyên dương phân biệt u_i và v_i mô tả một đoạn dây kết nối trực tiếp hai tấm thiệp u_i và v_i trên cây ($1 \leq u_i, v_i \leq N$).
- Dòng thứ t trong số Q dòng tiếp theo chứa ba số nguyên dương x_t, y_t, w_t mô tả lần thực hiện phép thuật tại thời điểm t của ông già Noel ($1 \leq x_t, y_t \leq N; w_t \leq 2 \times 10^5$).

Các số trên cùng một dòng cách nhau bởi dấu cách.

Kết quả

Ghi ra file văn bản PINE.OUT:

- Gồm Q dòng, dòng thứ t chứa một số nguyên là độ đẹp của cây sau khi thực hiện phép thuật tại thời điểm t .

Ví dụ

PINE.INP	PINE.OUT	Giải thích
5 3	0	- Ở thời điểm $t = 1$, dãy A là $(2, 1, 4, 4, 3)$. Kết quả là $(2 \% 1) + (1 \% 1) + (4 \% 1) + (4 \% 1) + (3 \% 1) = 0$.
2 1 4 4 8	3	
1 2	4	- Ở thời điểm $t = 2$, dãy A là $(2, 1, 1, 1, 0)$. Kết quả là $(2 \% 2) + (1 \% 2) + (1 \% 2) + (1 \% 2) + (0 \% 2) = 3$.
2 3		
3 4		- Ở thời điểm $t = 3$, dãy A là $(2, 1, 0, 1, 0)$. Kết quả là $(2 \% 3) + (1 \% 3) + (0 \% 3) + (1 \% 3) + (0 \% 3) = 4$.
4 5		
1 5 5		
1 5 3		
3 3 1		

Chấm điểm

- Subtask 1 (20% số điểm): $N, Q \leq 5\,000$.
- Subtask 2 (20% số điểm): $A_v \leq 2$ với mọi $v = 1, 2, \dots, N$; có dây nối trực tiếp giữa tấm thiệp v và $v + 1$ với mọi $v = 1, 2, \dots, N - 1$.
- Subtask 3 (20% số điểm): $x_t = y_t$ với mọi $t = 1, 2, \dots, Q$; có dây nối trực tiếp giữa tấm thiệp v và $v + 1$ với mọi $v = 1, 2, \dots, N - 1$.
- Subtask 4 (20% số điểm): Có dây nối trực tiếp giữa tấm thiệp v và $v + 1$ với mọi $v = 1, 2, \dots, N - 1$.
- Subtask 5 (20% số điểm): Không có ràng buộc nào thêm.

Bài 6. Cắm trại (6,0 điểm)

Vào năm mới, K gia đình tổ chức cắm trại trên một khu đất có dạng hình tròn. Khu đất được biểu diễn trên hệ trục tọa độ Đề-các Oxy (đơn vị mét), có tâm ở gốc tọa độ $O(0, 0)$ và bán kính $R = 100$ mét. Trên biên hình tròn, chọn K điểm phân biệt để làm vị trí cắm trại cho K gia đình. Các điểm được đánh số từ 1 đến K và được xác định bằng dãy góc $\alpha_1, \alpha_2, \dots, \alpha_K$ ($0^\circ \leq \alpha_1 < \alpha_2 < \dots < \alpha_K \leq 359^\circ$), α_i là góc từ tia Ox ngược chiều kim đồng hồ tới tia nối O đến điểm thứ i . Cụ thể, điểm thứ i ($1 \leq i \leq K$) có tọa độ là $(R \times \cos \frac{\alpha_i \times \pi}{180^\circ}, R \times \sin \frac{\alpha_i \times \pi}{180^\circ})$, trong bài toán này lấy $\pi = 3.14159265359$.

Alice nhận nhiệm vụ thiết kế đường dây để cấp điện cho tất cả các trại. Cụ thể, máy phát điện đặt ở trại 1, cần kết nối trại 1 tới $K - 1$ trại còn lại bằng dây điện. Trong quá trình thiết kế, được phép thêm tối đa 25 điểm trung gian để đấu nối nhưng các điểm này phải nằm trong hoặc trên biên của mảnh đất hình tròn. Các dây điện có thể đấu nối giữa hai trại, giữa một trại với một điểm trung gian, giữa hai điểm trung gian. Độ dài dây điện nối trực tiếp giữa hai điểm có tọa độ (x_1, y_1) và (x_2, y_2) là $\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$. Cần thiết kế sao cho với một trại bất kỳ, tồn tại duy nhất một đường dẫn điện trực tiếp hoặc gián tiếp từ trại 1 tới trại đó. Giả sử phần dây hao hụt trong quá trình đấu nối là không đáng kể.

Alice đã thiết kế được một phương án có tổng độ dài dây điện sử dụng là A mét. Cô đã mua đúng A mét dây điện. Tuy nhiên chưa kịp triển khai thì bản vẽ bị mất. Alice nhờ Bob thiết kế lại một phương án, nếu phương án của Bob có tổng độ dài dây điện cần dùng lớn hơn A thì họ phải mua thêm cho đủ, ngược lại thì không cần mua thêm.

Yêu cầu: Hãy giúp Bob tìm cách thiết kế sao cho tổng độ dài dây điện cần mua thêm là càng nhỏ càng tốt.

Dữ liệu

Vào từ file văn bản CAMP.INP:

- Dòng đầu chứa một số nguyên dương K ($3 \leq K \leq 10$).
- Dòng thứ hai chứa K số nguyên không âm $\alpha_1, \alpha_2, \dots, \alpha_K$.
- Dòng cuối cùng ghi số thực A ($0 < A < 2 \times \pi \times R$).

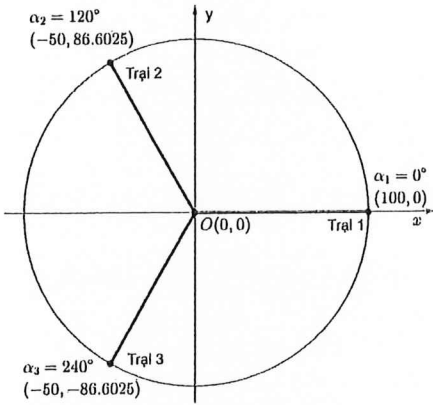
Các số trên cùng một dòng cách nhau bởi dấu cách.

Kết quả

Ghi ra file văn bản CAMP.OUT:

- Dòng đầu ghi một số tự nhiên M là số lượng điểm trung gian ($0 \leq M \leq 25$).
- Nếu $M > 0$ thì M điểm này được đánh số thứ tự từ $K + 1$ đến $K + M$, và ghi ra M dòng tiếp theo, dòng thứ i chứa hai số thực là toạ độ của điểm thứ $K + i$. Các giá trị toạ độ được ghi ra với độ chính xác đến bốn chữ số sau dấu chấm thập phân.
- Mỗi dòng trong số $K + M - 1$ dòng tiếp theo ghi hai số nguyên dương u và v , mô tả một dây điện nối giữa điểm thứ u và điểm thứ v ($1 \leq u, v \leq K + M$).

Ví dụ

CAMP . INP	CAMP . OUT	Minh hoạ
3 0 120 240 300.0000	1 0.0000 0.0000 1 4 2 4 3 4	Thêm một điểm trung gian có toạ độ $(0,0)$. Điểm trung gian này nối với 3 trại với tổng độ dài là 300.0000 mét. 

Chấm điểm

Với mỗi test, gọi B là tổng độ dài dây điện trong phương án thiết kế của Bob. Số điểm (trên thang điểm 1) cho phương án này là:

$$\begin{cases} 1 & \text{nếu } B \leq A \\ 0 & \text{nếu } B > 1.03 \times A \\ 13 - 12 \frac{B}{A} & \text{nếu } A < B \leq 1.03 \times A \end{cases}$$

Subtask	% số điểm	Giá trị K	Dãy α	Giá trị A
1	12%	3	(0, 90, 180)	273.2056
2	14%	3	(0, 40, 150)	230.5843
3	14%	4	(40, 140, 200, 340)	341.1483
4	14%	5	(0, 72, 144, 216, 288)	457.4330
5	6%	6	(0, 60, 120, 180, 240, 300)	500.0000
6	40%	Không có ràng buộc nào thêm		

----- HẾT -----

* Thí sinh **KHÔNG** được sử dụng tài liệu;

* Giám thị **KHÔNG** giải thích gì thêm.

ĐÁP ÁN
ĐỀ THI CHÍNH THỨC

Môn: TIN HỌC

Ngày thi: 25-26/12/2025

(Đáp án gồm 09 trang, 06 bài)

TỔNG QUAN ĐỀ THI

	Tiêu đề	File nộp	File dữ liệu	File kết quả	Thời gian	Bộ nhớ
Bài 1	Dãy đèn	LIGHT.*	LIGHT.INP	LIGHT.OUT	1 giây	1024 MB
Bài 2	Quà Noel	GIFT.*	GIFT.INP	GIFT.OUT	1 giây	1024 MB
Bài 3	Bài đăng	POST.*	POST.INP	POST.OUT	1 giây	1024 MB
Bài 4	Tất niên	TET.*	TET.INP	TET.OUT	1 giây	1024 MB
Bài 5	Cây thông	PINE.*	PINE.INP	PINE.OUT	1 giây	1024 MB
Bài 6	Cắm trại	CAMP.*	CAMP.INP	CAMP.OUT	3 giây	1024 MB

- Dấu * được thay thế bởi PAS hoặc CPP hoặc PY tương ứng với ngôn ngữ lập trình Pascal hoặc C++ hoặc Python.
- Mỗi bài bao gồm nhiều subtask, mỗi subtask bao gồm nhiều test đơn, điểm của thí sinh được tính theo từng test đơn.
- Thí sinh không được phép sử dụng các định hướng biên dịch chương trình có các từ khoá sau: pragma, optimize, target, O3, Ofast, unroll-loops, avx, avx2, fma, omit-frame-pointer.
- Bài của thí sinh sẽ được chấm điểm trên máy chấm có CPU tương đương với Intel Core i9-13900K 3.20 GHz. Thí sinh sử dụng thông tin máy chấm để tham khảo và đánh giá thời gian chạy đáp ứng yêu cầu đề bài.

Bài 1. Dây đèn (7,0 điểm)

Dữ liệu

Vào từ file văn bản LIGHT.INP:

- Dòng đầu chứa 4 số nguyên không âm N, Q, X, Y ($0 < X + Y \leq 5; X + Y \leq N \leq 666; 1 \leq Q \leq 10^5$).
- Dòng thứ hai chứa N số nguyên, số thứ i có giá trị 0, 1 hoặc 2 tương ứng với trạng thái ban đầu của bóng đèn thứ i là tắt, vàng hoặc đỏ ($1 \leq i \leq N$).
- Mỗi dòng trong số Q dòng tiếp theo chứa hai số nguyên dương L và R mô tả một yêu cầu. Dữ liệu bảo đảm $R - L + 1 \geq X + Y$ trong tất cả các yêu cầu khảo sát.

Kết quả

Ghi ra file văn bản LIGHT.OUT:

- Gồm Q dòng, mỗi dòng một số nguyên là số lượng thao tác ít nhất tìm được cho yêu cầu khảo sát tương ứng hoặc -1 nếu không tồn tại phương án.

Chấm điểm

- Subtask 1 (30% số điểm): $X = Y = 1; N \leq 10; Q = 1$.
- Subtask 2 (20% số điểm): $X = Y = 1; Q = 1$.
- Subtask 3 (34% số điểm): $Q \leq 5$.
- Subtask 4 (16% số điểm): Không có ràng buộc nào thêm.

Hướng dẫn giải chi tiết

- **Quan sát trạng thái.** Với một đoạn có c_0, c_1, c_2 đèn ở trạng thái tắt, vàng, đỏ ($c_0 + c_1 + c_2 = \ell$), một thao tác chọn X đèn quay theo chiều kim đồng hồ và Y đèn quay ngược chiều kim đồng hồ chỉ thay đổi số lượng của ba loại theo công thức

$$\begin{aligned}c'_0 &= c_0 - x_0 - y_0 + x_1 + y_2, \\c'_1 &= c_1 - x_1 - y_1 + x_2 + y_0, \\c'_2 &= c_2 - x_2 - y_2 + x_0 + y_1,\end{aligned}$$

trong đó x_i ($0 \leq i \leq 2$) là số đèn màu i được quay cw và y_i là số đèn màu i được quay ccw, với:

$$\sum_{i=0}^2 x_i = X, \quad \sum_{i=0}^2 y_i = Y, \quad 0 \leq x_i \leq c_i, \quad 0 \leq y_i \leq c_i - x_i.$$

Vì $X + Y \leq 5$, số cách phân bố (x_i, y_i) là hằng số và có thể liệt kê trực tiếp (tối đa 60 tổ hợp).

- **Tiền xử lý bằng BFS trên không gian trạng thái.** Ta xây dựng đồ thị có các đỉnh là các bộ ba (c_0, c_1, c_2) với $0 \leq c_0 \leq M$, $0 \leq c_1, c_2 \leq N$ và $c_0 + c_1 + c_2 \leq N$ (ở đây $M = 6$ vì với $c_0 > 6$ thì việc giảm c_0 đi một không ảnh hưởng kết quả). Bắt đầu BFS từ các trạng thái “tất cả tắt” $(i, 0, 0)$, $0 \leq i \leq N$. Ở mỗi đỉnh hiện tại ta duyệt tất cả các phân bố (x_i, y_i) hợp lệ, tính trạng thái kế tiếp (c'_0, c'_1, c'_2) và cập nhật khoảng cách ngắn nhất.
- **Truy vấn nhanh nhờ tiền tố.** Ta tính ba dãy cộng dồn để lưu số lượng đèn từng màu trong các đoạn tiền tố. Khi đó có thể tính nhanh trạng thái của truy vấn.
- **Độ phức tạp.** BFS duyệt tối đa $O(M \cdot N^2)$ trạng thái ($M = 6, N \leq 666$) và mỗi trạng thái xét không quá 60 cách chuyển trạng thái và bộ nhớ $O(M \cdot N^2)$. Mỗi truy vấn chỉ cần $O(1)$ thời gian nhờ tiền tố, tổng thời gian cho $Q \leq 10^5$ truy vấn là $O(Q)$.

Bài 2. Quà Noel (7,0 điểm)

Dữ liệu

Vào từ file văn bản GIFT.INP:

- Dòng đầu chứa 2 số nguyên N, Q ($1 \leq N, Q \leq 10^5$).
- Dòng thứ hai chứa N số nguyên dương $W_1, W_2, \dots, W_N$ có giá trị không quá 10^5 .
- Dòng thứ ba chứa N số nguyên dương $S_1, S_2, \dots, S_N$ có giá trị không quá 10^5 .
- Mỗi dòng trong số Q dòng tiếp theo thể hiện một câu hỏi thuộc một trong hai loại:
 - Loại 1, gồm bốn số nguyên dương: $1 \ M \ K \ T$ ($1 \leq M, K, T \leq 10^9$).
 - Loại 2, gồm ba số nguyên dương: $2 \ M \ K$ ($1 \leq M, K \leq 10^9$).

Các số trên cùng một dòng cách nhau bởi dấu cách.

Kết quả

Ghi ra file văn bản GIFT.OUT:

- Gồm Q dòng, mỗi dòng thể hiện câu trả lời cho một câu hỏi tương ứng:
 - Loại 1: Ghi ra 1 nếu có thể tạo được T túi quà, trái lại ghi ra 0.
 - Loại 2: Ghi ra một số nguyên là số lượng túi quà nhiều nhất tạo được.

Chấm điểm

- Subtask 1 (25% số điểm): $S_i = 1, \forall i = 1, 2, \dots, N$.
- Subtask 2 (20% số điểm): $N, Q \leq 1\,000$ và chỉ có câu hỏi loại 1.
- Subtask 3 (15% số điểm): $N, Q \leq 1\,000$.
- Subtask 4 (20% số điểm): Chỉ có câu hỏi loại 1.
- Subtask 5 (20% số điểm): Không có ràng buộc nào thêm.

Hướng dẫn giải chi tiết

- **Quan sát:** Để tạo x túi, mỗi túi có đúng K món khác loại, cần $x \times K$ món. Với loại i có S_i món, khi tạo x túi ta có thể dùng tối đa $\min(S_i, x)$ món của loại này. Vì mục tiêu là giảm tổng khối lượng, nên luôn chọn các món có trọng lượng W_i nhỏ nhất; do đó chi phí tối thiểu cho x túi được xác định khi lấy $x \times K$ món nhẹ nhất trong tập các món “cắt” thành $\min(S_i, x)$ phần.
- **Yêu cầu bài toán:**
 - Loại 1: cho M, K, T cần kiểm tra xem tồn tại $x = T$ sao cho chi phí tối thiểu của x túi $\leq M$.

- Loại 2: cho M, K cần tìm $x_{\max}$ sao cho chi phí tối thiểu của $x_{\max}$ túi $\leq M$.

Cả hai dạng đều quy về việc tìm giá trị lớn nhất x (số túi) thỏa mãn điều kiện chi phí $\leq M$.

- **Thuật toán:** Áp dụng phương pháp *chia để trị trên câu trả lời* (tìm kiếm nhị phân trên x) đồng thời xử lý tất cả truy vấn bằng các cây Fenwick (BIT):

- Số loại có $S_i > x$ (có thể cung cấp x món mỗi loại).
- Tổng số món đã dùng ở các loại có $S_i \leq x$.
- Tổng khối lượng của các món ở nhóm $S_i \leq x$.
- Tổng trọng lượng W_i của các loại $S_i > x$ (mỗi loại còn x món chưa dùng).

Các loại được sắp xếp tăng dần theo S_i ; khi chuyển từ khoảng $[L, R)$ sang mid ta cập nhật BIT cho những loại có $S_i \leq mid$. Đối với mỗi truy vấn (M, K) ta dùng “find-first” trên BIT để lấy nhanh số món còn thiếu và chi phí hiện tại, so sánh với M để đưa truy vấn sang nửa trái (không đủ) hoặc nửa phải (đủ). Cuối cùng giá trị trả về cho truy vấn là L (độ lớn nhất thỏa mãn).

- **Độ phức tạp:**

- Sắp xếp các loại: $O(N \log N)$.
- Mỗi loại được cập nhật và phục hồi trong một lần chia: $O(\log N)$, tổng $O(N \log N)$.
- Mỗi truy vấn tham gia vào quá trình chia $O(\log N)$ lần, mỗi lần thực hiện một truy vấn BIT với $O(\log N)$, nên tổng $O(Q \log^2 N)$.

Bài 3. Bài đăng (6,0 điểm)

Chăm điểm

- Subtask 1 (15% số điểm): $N, Q \leq 50$.
- Subtask 2 (15% số điểm): $N \leq 500$.
- Subtask 3 (10% số điểm): $N \leq 5\,000$.
- Subtask 4 (20% số điểm): $Q = 1; U = 1; V = N$.
- Subtask 5 (20% số điểm): $Q \leq 30\,000$.
- Subtask 6 (20% số điểm): Không có ràng buộc nào thêm.

Hướng dẫn giải chi tiết

- **Quan sát.** Với một đoạn $[L, R]$ là *toàn vẹn* hoặc $[L, R]$ là đoạn toàn vẹn ngắn nhất bắt đầu từ L , hoặc tồn tại M mà $[L, M]$ và $[M + 1, R]$ là đoạn toàn vẹn. Gọi $f[L]$ là số đoạn toàn vẹn bắt đầu tại L , $f[L] = f[M + 1] + 1$ với M là giá trị nhỏ nhất mà $[L, M]$ là đoạn toàn vẹn. Như vậy, bài toán đưa về tìm đoạn toàn vẹn nhỏ nhất bắt đầu từ L .

- Thuật toán.

1. Dùng hai ngăn xếp đơn điệu (đối với vị trí phải nhất và trái nhất) duyệt ngược từ $i = N$ tới 1 để tính mảng $\text{nxt}[i]$: là chỉ số j nhỏ nhất sao cho $[i, j]$ chứa đầy đủ mọi giá trị bắt đầu ở i (đây là đoạn *toàn vẹn* bắt đầu tại i).
2. Nhóm các truy vấn theo L và duyệt L giảm dần. Gọi $\text{cnt}[i]$ là số lượng toàn vẹn kết thúc tại i mà vị trí bắt đầu $\geq L$.
3. Khi đang ở L , cập nhật các đoạn *toàn vẹn* có biên trái L bằng cách đi dọc chuỗi các chỉ số $i = \text{nxt}[L], \text{nxt}[\text{nxt}[L] + 1], \dots$; đối với mỗi i ta tăng $\text{cnt}[i]$ lên 1, như vậy kết quả cho mỗi truy vấn là $\sum_{U \leq i \leq V} \text{cnt}[i]$. Tuy nhiên như vậy việc cập nhật sẽ tốn rất nhiều thời gian.
4. Để cải tiến, ta chia dãy các vị trí $1, 2, \dots, N$ thành $\sqrt{N}$ khối, khi đó với mỗi L , ta chỉ cần cập nhật tổng $\text{cnt}[i]$ cho mỗi khối và vị trí i đầu tiên của khối được tăng $\text{cnt}[i]$ lên 1. Dựa vào các thông tin này sẽ tính được kết quả cho mỗi truy vấn trong $O(\sqrt{N})$.

- Độ phức tạp.

- Độ phức tạp thời gian: $O(N)$ (tiền xử lý) + $O((N + Q)\sqrt{N})$ (truy vấn).
- Độ phức tạp bộ nhớ: $O(N)$.

Bài 4. Tất niên (7,0 điểm)

Chăm điểm

- Subtask 1 (30% số điểm): $Q = 1, N \leq 200$.
- Subtask 2 (30% số điểm): $Q = 1, N \leq 2000$.
- Subtask 3 (20% số điểm): $Q = 1$.
- Subtask 4 (20% số điểm): Không có ràng buộc nào thêm.

Hướng dẫn giải chi tiết

- Quan sát. Với một đoạn độ dài M và các giá trị $a_1, \dots, a_M$, số dãy khác nhau thu được bằng không quá một phép đảo ngược bằng

$$1 + \binom{M}{2} - \sum_v \binom{c_v}{2},$$

trong đó c_v là số lần xuất hiện của giá trị v trong đoạn. Công thức này xuất phát từ việc mỗi phép đảo ngược chỉ làm thay đổi thứ tự của một cặp vị trí $i < j$; nếu $a_i = a_j$ thì dãy sau đảo trùng với dãy trước, còn nếu $a_i \neq a_j$ thì tạo ra một dãy mới. Do đó số dãy mới bằng số cặp vị trí có giá trị khác nhau, tức $\binom{M}{2} - \sum_v \binom{c_v}{2}$, cộng với dãy gốc.

- **Thuật toán.** Xử lý các truy vấn ngược lại: bắt đầu từ trạng thái sau khi đã bỏ hết Q đĩa, rồi lần lượt “thêm lại” các đĩa theo thứ tự ngược của $p_1, \dots, p_Q$. Sau mỗi lần thêm, ta hợp nhất các đoạn liên tục hiện có (nếu đĩa mới kề với một hoặc hai đoạn) và cập nhật giá trị độ đa dạng của đoạn mới bằng một multiset. Giá trị đa dạng lớn nhất hiện tại luôn là phần tử lớn nhất của một multiset của tất cả các đoạn đang tồn tại.
- **Độ phức tạp.** Mỗi lần hợp nhất sẽ duyệt dãy có ít phần tử hơn để hợp nhất, và cần dùng cấu trúc dữ liệu để lưu lại số lượng của một giá trị xuất hiện. Vì vậy tổng thời gian là

$$O(N \log^2 N).$$

Bài 5. Cây thông (7,0 điểm)

Dữ liệu

Vào từ file văn bản PINE.INP:

- Dòng đầu chứa hai số nguyên dương N và Q ($1 \leq N, Q \leq 2 \times 10^5$).
- Dòng thứ hai chứa N số nguyên $A_1, A_2, \dots, A_N$ ($0 \leq A_i \leq 2 \times 10^5$).
- Dòng thứ i trong số $N - 1$ dòng tiếp theo chứa hai số nguyên dương phân biệt u_i và v_i mô tả một đoạn dây kết nối trực tiếp hai tấm thiệp u_i và v_i trên cây ($1 \leq u_i, v_i \leq N$).
- Dòng thứ t trong số Q dòng tiếp theo chứa ba số nguyên dương x_t, y_t, w_t mô tả lần thực hiện phép thuật tại thời điểm t của ông già Noel ($1 \leq x_t, y_t \leq N; w_t \leq 2 \times 10^5$).

Các số trên cùng một dòng cách nhau bởi dấu cách.

Kết quả

Ghi ra file văn bản PINE.OUT:

- Gồm Q dòng, dòng thứ t chứa một số nguyên là độ đẹp của cây sau khi thực hiện phép thuật tại thời điểm t .

Chấm điểm

- Subtask 1 (20% số điểm): $N, Q \leq 5\,000$.
- Subtask 2 (20% số điểm): $A_v \leq 2$ với mọi $v = 1, 2, \dots, N$; có dây nối trực tiếp giữa tấm thiệp v và $v + 1$ với mọi $v = 1, 2, \dots, N - 1$.
- Subtask 3 (20% số điểm): $x_t = y_t$ với mọi $t = 1, 2, \dots, Q$; có dây nối trực tiếp giữa tấm thiệp v và $v + 1$ với mọi $v = 1, 2, \dots, N - 1$.
- Subtask 4 (20% số điểm): Có dây nối trực tiếp giữa tấm thiệp v và $v + 1$ với mọi $v = 1, 2, \dots, N - 1$.
- Subtask 5 (20% số điểm): Không có ràng buộc nào thêm.

Hướng dẫn giải chi tiết

- **Quan sát:** Mỗi phép thuật chỉ thay đổi các đỉnh nằm trên một đường $x_t - y_t$ của cây bằng phép $A_v \leftarrow A_v \bmod w_t$. Các giá trị A_v hoặc là giữ nguyên (nếu $A_v < w_t$) hoặc là giảm đi ít nhất một nửa (nếu $A_v \geq w_t$). Do đó tổng số lần mà các A_v giảm giá trị là không quá $n \times \log(A_{\max})$. Truy vấn thứ t có thể thực hiện với độ phức tạp $O(\frac{A_{\max}}{t} \times \log(A_{\max}))$ khi biết tần suất xuất hiện của từng giá trị trong mảng A .
- **Cấu trúc dữ liệu:**
 1. **Heavy-Light Decomposition (HLD):** Đánh số lại các đỉnh theo quy tắc nhánh lớn hơn thăm trước, đưa mỗi đường $x_t - y_t$ thành $O(\log N)$ đoạn liên tiếp.
 2. **Segment tree (ST):** Quản lý mảng A với thao tác thay đổi A_v và tìm $\max(A_v)$ trên một đoạn liên tiếp.
 3. **Fenwick (BIT):** Dùng để đếm số lượng và tính tổng các giá trị trong mảng A (trên mảng tần suất), phục vụ cho bước trả lời truy vấn.
- **Thuật toán cho mỗi phép thuật t (đầu vào x_t, y_t, w_t):**
 1. Lấy danh sách các đoạn $[l, r]$ của đường $x_t - y_t$ nhờ HLD.
 2. Tìm các giá trị $A_v \geq w_t$ thuộc các đoạn, sử dụng ST.
 3. Với mỗi giá trị A_v , thực hiện thay đổi và cập nhật trên cả ST và BIT.
 4. Tính kết quả bằng việc quét các bội t trên BIT.
- **Độ phức tạp:**
 - Mỗi đỉnh chỉ giảm giá trị khi $A_v \geq M$; sau mỗi giảm giá trị nó ít nhất một lần bị chia cho một số ≥ 2 , do đó số lần thay đổi của một đỉnh $O(\log A_{\max})$.
 - Vì ST duyệt chỉ các nút có $\max \geq M$, tổng thời gian cập nhật trên toàn bộ Q truy vấn là $O((N + Q) \log N \log A_{\max})$.
 - Tính kết quả truy vấn thực hiện tổng cộng $\sum_{t=1}^Q (\frac{A_{\max}}{t})$ thao tác trên BIT, có tổng độ phức tạp $O(A_{\max} \log^2(A_{\max}))$.
 - Như vậy tổng thời gian toàn thuật toán là $O((N + Q) \log N \log A_{\max} + A_{\max} \log^2(A_{\max}))$.

Bài 6. Cắm trại (6,0 điểm)

Dữ liệu

Vào từ file văn bản CAMP.INP:

- Dòng đầu chứa một số nguyên dương K ($3 \leq K \leq 10$).
- Dòng thứ hai chứa K số nguyên không âm $\alpha_1, \alpha_2, \dots, \alpha_K$.
- Dòng cuối cùng ghi số thực A ($0 < A < 2 \times \pi \times R$).

Các số trên cùng một dòng cách nhau bởi dấu cách.

Kết quả

Ghi ra file văn bản CAMP.OUT:

- Dòng đầu ghi một số tự nhiên M là số lượng điểm trung gian ($0 \leq M \leq 25$).
- Nếu $M > 0$ thì M điểm này được đánh số thứ tự từ $K + 1$ đến $K + M$, và ghi ra M dòng tiếp theo, dòng thứ i chứa hai số thực là toạ độ của điểm thứ $K + i$. Các giá trị toạ độ được ghi ra với độ chính xác đến bốn chữ số sau dấu chấm thập phân.
- Mỗi dòng trong số $K + M - 1$ dòng tiếp theo ghi hai số nguyên dương u và v , mô tả một dây điện nối giữa điểm thứ u và điểm thứ v ($1 \leq u, v \leq K + M$).

Chấm điểm

Với mỗi test, gọi B là tổng độ dài dây điện trong phương án thiết kế của Bob. Số điểm (trên thang điểm 1) cho phương án này là:

$$\begin{cases} 1 & \text{nếu } B \leq A \\ 0 & \text{nếu } B > 1.03 \times A \\ 13 - 12 \frac{B}{A} & \text{nếu } A < B \leq 1.03 \times A \end{cases}$$

Subtask	% số điểm	Giá trị K	Dãy α	Giá trị A
1	12%	3	(0, 90, 180)	273.2051
2	14%	3	(0, 40, 150)	230.5843
3	14%	4	(40, 140, 200, 340)	341.1475
4	14%	5	(0, 72, 144, 216, 288)	457.4330
5	6%	6	(0, 60, 120, 180, 240, 300)	500.0000
6	40%	Không có ràng buộc nào thêm		

Hướng dẫn giải chi tiết

- **Quan sát.** Vấn đề yêu cầu xây dựng một mạng cây nối tất cả K trại (đỉnh đầu vào) sao cho tổng độ dài các dây điện B càng gần A càng tốt và không vượt quá 25 điểm trung gian. Khi không dùng điểm trung gian, tổng độ dài nhỏ nhất chính là tổng độ dài các cạnh của *cây khung tối thiểu* (MST) trên K điểm. Thêm các điểm trung gian chỉ làm giảm được độ dài so với MST nếu chúng được dùng làm *điểm Steiner*. Vì $K \leq 10$, ta có thể xét một tập hợp hữu hạn các vị trí khả thi cho các điểm Steiner (ví dụ một lưới đều trong vòng tròn) và chuyển bài toán sang *Steiner tree trên đồ thị*.
- **Thuật toán.**
 1. **Tạo đồ thị.**
 - Gọi $V_0 = K$ là các trại đã cho.
 - Xây dựng một lưới vuông đều bên trong vòng tròn bán kính $R = 100$ với khoảng cách Δ (độ mịn của lưới ảnh hưởng tới độ chính xác của nghiệm).
 - Gộp các điểm này với V_0 thành tập đỉnh V .

– Nói mỗi cặp đỉnh $u, v \in V$ bằng một cạnh có trọng số $\text{dist}(u, v) = \sqrt{(x_u - x_v)^2 + (y_u - y_v)^2}$.

2. **DP Steiner tree.** Đặt $T = \{1, \dots, K\}$ là tập các trại. Định nghĩa $\text{dp}[S][v]$ là chiều dài ngắn nhất của một cây con nối tất cả các đầu mối trong tập con $S \subseteq T$ và có đỉnh v là đỉnh cuối (có thể là một điểm Steiner). Khởi tạo $\text{dp}[\{i\}][i] = 0$ và $\text{dp}[\{i\}][v] = +\infty$ với $v \neq i$. Đối với mỗi mask S (theo thứ tự tăng dần) và mỗi phân hoạch $S = S_1 \cup S_2$ (với $S_1 \cap S_2 = \emptyset$) ta thực hiện

$$\text{dp}[S][v] = \min(\text{dp}[S][v], \text{dp}[S_1][v] + \text{dp}[S_2][v]).$$

Sau khi đã gộp các sub-mask, chạy một vòng Dijkstra (hoặc SPFA) trên đồ thị để truyền thông tin “đi qua” các cạnh:

$$\text{dp}[S][u] \leftarrow \min(\text{dp}[S][u], \text{dp}[S][v] + \text{dist}(v, u)) \quad \forall (v, u) \in E.$$

3. **Khôi phục nghiệm.** Khi DP kết thúc, vị trí v^* cho $\min_v \text{dp}[T][v]$ cho tổng độ dài B nhỏ nhất đạt được. Dùng mảng trace ghi lại cách mà mỗi trạng thái được tạo ra (kết hợp sub-mask hay mở rộng qua một cạnh) để dựng lại tập các đỉnh và các cạnh thuộc cây Steiner tối ưu. Các đỉnh không thuộc tập gốc V_0 được liệt kê ra là các điểm trung gian M . Cuối cùng xuất ra các cạnh dưới dạng cặp chỉ số (đánh số lại các đỉnh theo thứ tự: trại $1 \dots K$, rồi các điểm trung gian $K+1 \dots K+M$).
4. **Đánh giá độ dài so với A .** Nếu $B \leq A$ thì Bob không cần mua thêm dây (kết quả đạt mục tiêu). Nếu $B > A$ thì $B - A$ là lượng dây phải mua thêm; do DP đã cho B là nhỏ nhất có thể trong không gian các điểm trung gian đã xét, nên $B - A$ là giá trị tối ưu (hoặc gần tối ưu nếu lưới chưa đủ mịn). Độ mịn lưới có thể tăng lên để cải thiện B nếu cần.
5. **Chia lần hai.** Các điểm được bổ sung ở lần chia thứ nhất có thể được điều chỉnh qua vị trí lân cận, bằng cách chia không gian xung quanh các điểm đó thành lưới, sau đó DP tương tự như lần chia thứ nhất. Lần chia lưới thứ nhất giúp tìm ra “hình dạng” của nghiệm, trong khi lần chia thứ hai giúp nghiệm trở nên “mịn” hơn.

----- HẾT -----