



ĐỀ CHÍNH THỨC

Ngày thi: 04 tháng 8 năm 2023

Thời gian làm bài: 180 phút (không kể thời gian giao đề)

Đề thi gồm có 03 trang

Họ và tên thí sinh: Số báo danh:

TỔNG QUAN ĐỀ THI

Bài	Tên bài	Tập chương trình	Tập dữ liệu	Tập kết quả	Điểm
1	ĐOẠN NGUYÊN TỐ	TPRIME.*	TPRIME.INP	TPRIME.OUT	100
2	HÀNG CÂY	PROLINE.*	PROLINE.INP	PROLINE.OUT	100
3	TẮC ĐƯỜNG	JAMONE.*	JAMONE.INP	JAMONE.OUT	100

Dấu * được thay thế bởi pas hoặc cpp của ngôn ngữ lập trình sử dụng tương ứng là Pascal hoặc C++.

BÀI 1. ĐOẠN NGUYÊN TỐ

Cho một dãy số nguyên dương $A = (a_1, a_2, \dots, a_n)$ ($a_i \leq 10^6; 1 \leq i \leq n$). Với mỗi phần tử a_i bạn được phép tăng hoặc giảm một lượng tùy ý để được một số nguyên tố. Khi đó chi phí của bạn cần bỏ ra chính là lượng tăng hoặc giảm đó.

Yêu cầu: Hãy chọn ra một đoạn con gồm k phần tử liên tiếp nhau của dãy A sao cho tổng chi phí biến đổi mọi phần tử trong đoạn con đó thành các số nguyên tố là nhỏ nhất.

Dữ liệu: Vào từ file TPRIME.INP có cấu trúc như sau: [Trại hè Hùng Vương Confessions - fb.com/cfs.traihehungvuong](https://fb.com/cfs.traihehungvuong)

- Dòng 1: Chứa hai số nguyên dương n, k ($1 \leq k \leq n \leq 10^5$);
- Dòng 2: Chứa n số nguyên dương a_1, a_2, \dots, a_n ($a_i \leq 10^6 \forall i = 1, 2, \dots, n$).

Kết quả: Ghi ra file TPRIME.OUT một số nguyên duy nhất là tổng chi phí biến đổi nhỏ nhất tìm được.

Ví dụ:

TPRIME.INP	TPRIME.OUT
4 2 9 5 8 15	1

Giải thích: chọn đoạn $[5, 8]$, biến đổi $8 \rightarrow 7$ với chi phí là 1.

Ràng buộc:

- 20% số test tương ứng với 20% số điểm có a_i đều là số nguyên tố $\forall i = 1, 2, \dots, n$;
- 40% test khác tương ứng với 40% số điểm có $n, k \leq 5000; a_i \leq 5000 \forall i = 1, 2, \dots, n$;
- 40% test còn lại ứng với 40% số điểm không có ràng buộc bổ sung.

BÀI 2. HÀNG CÂY

Trên một tuyến đường có n cây cổ thụ thuộc một trong n loại khác nhau. Từ đầu đến cuối đoạn đường, cây thứ i có chiều cao là h_i . Chính quyền thành phố muốn loại bỏ một số cây để những cây còn lại có chiều cao tăng dần từ đầu đến cuối đoạn đường. Đồng thời, để tăng tính thẩm mỹ, lãnh đạo cũng muốn số cây còn lại không nhỏ hơn 3 và chiều cao của các cây giữ lại nhìn từ đầu tới cuối tạo thành một cấp số cộng. Nói cách khác, giả sử giữ lại k cây, và các cây được giữ lại có chỉ số p_1, p_2, \dots, p_k ($1 \leq p_1 < p_2 < \dots < p_k \leq n; k \geq 3$) thì:

$$h_{p_2} - h_{p_1} = h_{p_3} - h_{p_2} = \dots = h_{p_k} - h_{p_{k-1}} > 0.$$

An đang thắc mắc liệu có bao nhiêu cách chọn cây thỏa mãn yêu cầu của lãnh đạo chính quyền thành phố đưa ra. Hai cách chọn được gọi là khác nhau nếu tồn tại một cây được chọn ở cách này nhưng không xuất hiện ở cách chọn còn lại.

Dữ liệu: Vào từ file **PROLINE.INP**

- Dòng đầu chứa số nguyên dương n ($n \leq 10^4$);
- Dòng thứ hai chứa n số nguyên dương h_1, h_2, \dots, h_n ($h_i \leq 10^6 \forall i = 1, 2, \dots, n$).

Kết quả: Ghi ra file **PROLINE.OUT** một số nguyên là số cách tìm được lấy theo module 1000000007.

PROLINE.INP	PROLINE.OUT
6	5
3 1 3 5 9 7	Trại hè Hùng Vương Confessions - fb.com/cfs.traihehungvuong

Giải thích: Có 5 cách lựa chọn giữ lại các cây với thứ tự $\{1, 4, 6\}$; $\{2, 3, 4\}$; $\{2, 3, 4, 6\}$; $\{2, 4, 5\}$; $\{3, 4, 6\}$.

Ràng buộc:

- Có 15% số test tương ứng 15% số điểm có $n \leq 20$; $h_i = i; \forall i = 1, 2, \dots, n$;
- Có 15% số test tương ứng 15% số điểm có $n \leq 20$;
- Có 20% số test tương ứng 20% số điểm có $20 < n \leq 500$;
- Có 20% số test tương ứng 20% số điểm có $500 < n \leq 2000$;
- 30% số test còn lại có $2000 < n \leq 10^4$.

BÀI 3. TẮC ĐƯỜNG

VP là một thành phố vô cùng đặc biệt. Ngoài vẻ đẹp tự nhiên thu hút khách du lịch, hệ thống giao thông cũng vô cùng đặc biệt. Thành phố được kết nối bởi n điểm với m con đường phục vụ di chuyển đảm bảo liên thông toàn thành phố. Con đường thứ i kết nối hai chiều giữa hai điểm u_i và v_i với thời gian di chuyển là p_i . Tuy nhiên, trong trường hợp xấu xảy ra tắc đường, để đi hết con đường này cần thời gian là q_i .

Hà là một quản lý trong một công ty vận tải, Hà đang thiết kế lộ trình cố định cho tuyến xe phục vụ đưa đón khách du lịch bắt đầu từ s , di chuyển qua một số con đường và kết thúc tại điểm t . Giả thiết trong trường hợp xấu nhất, chỉ có một con đường trên tuyến đường di chuyển đó gặp sự cố tắc đường.

Yêu cầu: Cho k truy vấn, mỗi truy vấn gồm 2 số s, t ($1 \leq s, t \leq n$). Với mỗi truy vấn, hãy giúp Hà lựa chọn một lộ trình cố định sao cho trong trường hợp xấu nhất, tổng thời gian di chuyển của xe là nhỏ nhất.

Dữ liệu: Vào từ file **JAMONE.INP**

- Dòng đầu chứa 3 số nguyên n, m, k ;
- m dòng tiếp theo, dòng thứ i chứa 4 số nguyên dương u_i, v_i, p_i, q_i ($u_i, v_i \leq n; p_i < q_i \leq 10^5 \forall i = 1, 2, \dots, m$);
- k dòng cuối cùng, dòng thứ j chứa hai số nguyên dương s, t xác định thông tin trong truy vấn thứ j ($s, t \leq n; \forall j = 1, 2, \dots, k$).

Kết quả: Ghi ra file **JAMONE.OUT** gồm k dòng, dòng thứ j đưa ra câu trả lời cho truy vấn thứ j là tổng thời gian di chuyển theo tuyến đường Hà lựa chọn trong trường hợp xấu nhất.

Ví dụ:

JAMONE . INP	JAMONE . OUT	MINH HỌA
4 5 3 1 2 2 3 1 3 8 10 1 4 3 4 3 4 4 6 2 3 1 12 1 3 2 4 2 3	9 6 11	

Trại hè Hùng Vương Confessions - fb.com/cfs.traihehungvuong

Giải thích:

- Để di chuyển từ 1 tới 3 ta có 3 cách đi.
 - o Với cách đi $1 \rightarrow 3$ trong trường hợp xảy ra tắc thì thời gian di chuyển là 10.
 - o Với cách đi $1 \rightarrow 2 \rightarrow 3$ trong trường hợp xảy ra tắc ở đường $1 \rightarrow 2$ thì thời gian di chuyển là $3 + 1 = 4$ nhưng trong trường hợp xấu tắc ở $2 \rightarrow 3$ thì thời gian di chuyển là $2 + 12 = 14$.
 - o Với cách di chuyển $1 \rightarrow 4 \rightarrow 3$, trong trường hợp xảy ra tắc ở đường $1 \rightarrow 4$ thì thời gian di chuyển là $4 + 4 = 8$, trong trường hợp xấu tắc ở $4 \rightarrow 3$ thì thời gian di chuyển là $3 + 6 = 9$.

Ràng buộc:

- 20% số test tương ứng 20% số điểm có $n \leq 10; m \leq 20; k = 1$
- 20% số test khác tương ứng 20% số điểm có $n \leq 100; m \leq 1000; k \leq 10$; và
$$q_1 - p_1 = q_2 - p_2 = q_3 - p_3 = \dots = q_m - p_m$$
- 30% số test khác tương ứng 30% số điểm có $n \leq 300; m \leq 1000; k \leq 10$
- 30% số test còn lại có $n \leq 1000; m \leq 5000; k \leq 10$.

----- HẾT -----

*Thí sinh không sử dụng tài liệu.
Cán bộ coi thi không giải thích gì thêm.*



HƯỚNG DẪN GIẢI

Trại hè Hùng Vương Confessions - fb.com/cfs.traihehungvuong

BÀI 1: ĐOẠN NGUYÊN TỐ

Subtask 1: Có a_i đều là số nguyên tố $\forall i = 1, 2, \dots, n$;

Chỉ việc cout $<< 0$;

Subtask 2: $n, k \leq 5000$; $a_i \leq 5000$; $\forall i = 1, 2, \dots, n$;

- Với mỗi giá trị a_i ta duyệt trâu để tìm số nguyên tố gần a_i nhất \Rightarrow tính chi phí b_i .
- Duyệt trên mảng b tìm đoạn độ dài k có tổng nhỏ nhất.

Subtask 3 : $n, k \leq 10^5$; $a_i \leq 10^6$; $\forall i = 1, 2, \dots, n$;

- Sàng nguyên tố.
- Với mỗi giá trị a_i để tìm số nguyên tố nhỏ nhất ta có thể duyệt trực tiếp (do mật độ số nguyên tố tương đối dày) hoặc có thể dùng mảng $pre[i]$ và $next[i]$ cho biết số nguyên tố lớn nhất nhỏ hơn hoặc bằng i và số nguyên tố nhỏ nhất nhưng lớn hơn hoặc bằng i . Từ đó tính được chi phí b_i .
- Việc tìm đoạn độ dài k có tổng nhỏ nhất ta sử dụng mảng cộng dồn trên b .

BÀI 2: HÀNG CÂY

Subtask 1: $n \leq 20$; $h_i = i$; $\forall i = 1, 2, \dots, n$;

Duyệt các hiệu (công sai), ta tìm được số phần tử tối đa. Dãy cấp số cộng là một đoạn con liên tiếp.

Subtask 2: $n \leq 20$

Trại hè Hùng Vương Confessions - fb.com/cfs.traihehungvuong

Ta duyệt đệ quy dãy 0, 1 độ dài n để xác định việc chặt bỏ hay để lại các cây.

Subtask 3: $n \leq 500$

- Gọi $F[i][j]$ là số dãy cấp số cộng có hai phần tử cuối cùng là a_i và a_j ($a_i < a_j$).
- Ta có $F[i][j] = \sum F[k][i]$ với $k = 1 \dots i - 1$ và $a_i - a_k = a_j - a_i$. Duyệt k .

ĐPT $O(n^3)$

Subtask 4: : $n \leq 2000$ Quy hoạch động như subtask 3, dùng cấu trúc dữ liệu để tìm, tính nhanh $\sum F[k][i]$ dựa vào hiệu $a_i - a_k$. ĐPT $O(n^2 \log_2 n)$

Subtask 4: : $n \leq 10^4$

- Sắp xếp dãy tăng dần, ghi nhận vị trí cũ (pair)

- Ta thực hiện QHD như sub 3. Khi đó nhận thấy các vị trí k thỏa mãn sẽ đứng cạnh nhau, (trong đó có một số cây phía sau không được tính). Để tính tổng $\sum F[k][i]$ dùng công thức tính tổng tiền tố. Tổng độ phức tạp $O(n^2)$

BÀI 3: TẮC ĐƯỜNG

Subtask 1: $n \leq 10; m \leq 20; k = 1$

- Duyệt $i = 1 \dots m$ là cạnh xảy ra tắc đường.
- Ta xây dựng đồ thị mới với trọng số cạnh thứ j bằng $p_j \forall j \neq i$; riêng cạnh thứ i có trọng số là q_i .
- Tìm đường đi ngắn nhất từ s tới t .

Subtask 2: $n \leq 100; m \leq 1000; k \leq 10; q_1 - p_1 = q_2 - p_2 = q_3 - p_3 = \dots = q_m - p_m$

- Dùng thuật toán Floyd để tìm độ dài đường đi ngắn nhất giữa mọi cặp đỉnh với trọng số cạnh thứ i là p_i .
- Với mỗi truy vấn ta chỉ việc cộng thêm một đại lượng không đổi $q_1 - p_1$

Subtask 3: $n \leq 100; m \leq 1000; k \leq 10$

Trại hè Hùng Vương Confessions - fb.com/cfs.traihehungvuong

- Duyệt các cạnh theo thứ tự tăng dần về chênh lệch $q_i - p_i$
- Ta xây dựng đồ thị bằng cách thêm dần các cạnh theo thứ tự chênh lệch, khi đó kết quả sẽ là đường đi ngắn nhất cộng với chênh lệch $(q-p)$ của cạnh vừa thêm. Sử dụng thuật toán Dijkstra để cài đặt đường đi ngắn nhất. ĐPT $O(k \times m^2 \log_2 n)$

Subtask 4: $n \leq 1000; m \leq 5000; k \leq 10$

- Thực hiện như subtask 3, nhưng mỗi khi ta thêm cạnh vào, không thực hiện tìm đường đi ngắn nhất lại từ đầu. Ta chỉ thực DFS cập nhật tiếp theo những giá trị cập nhật được theo cạnh vừa thêm hoặc cài đặt như thuật toán Bellman-Ford. ĐPT $O(m \times (m + n))$