

ĐỀ THI CHÍNH THỨC

Đề thi có 07 trang

Môn: TIN HỌC

Thời gian: 300 phút (không kể thời gian giao đề)

Ngày thi thứ hai: 31/3/2018

TỔNG QUAN ĐỀ THI NGÀY THỨ HAI

Bài	Tên bài	Tên file chương trình	Hạn chế bộ nhớ
5	Tìm kiếm nhị phân	BSEARCH.*	512 M
6	Diện tích bao lồi	CONVEX.*	512 M
7	Bảo vệ thành phố	CITYDEF.*	512 M
8	Trò chơi trên đa giác	POLYGAME.*	512 M

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

Dữ liệu vào từ thiết bị vào chuẩn. Kết quả ghi ra thiết bị ra chuẩn.

Lập trình giải các bài toán sau đây:

Bài 5. (5 điểm) Tìm kiếm nhị phân

Vinh đang tích cực chuẩn bị để tham gia vào kỳ thi Olympic Tin học sinh viên Việt Nam, khối Siêu cúp. Chỉ còn không đến một tuần nữa là đến kỳ thi. Khi xem lại đề thi của những kỳ thi gần đây, Vinh nhận thấy còn chưa có nhiều bài liên quan đến việc áp dụng tìm kiếm nhị phân. Là người có nhiều kinh nghiệm từ các kỳ thi học sinh giỏi tin học quốc gia, Vinh nhanh chóng viết được đoạn mã cài đặt tìm kiếm nhị phân trên C++ sau đây:

```
int binary_search(vector<int> &a, int key) {
    int L = 0;
    int R = (int)a.size() - 1;
    while (L <= R) {
        int m = (L + R + 1) / 2;
        if (a[m] == key)
            return m;
        else if (a[m] > key)
            R = m - 1;
        else
            L = m + 1;
    }
    return -1;
}
```

Trước khi thực hiện kiểm thử chương trình, Vinh chợt nhận thấy rằng vector nguyên đầu vào là chưa được sắp xếp. Rất may là Vinh đã sớm phát hiện ra điều này để kịp thời xử lý. Tuy nhiên, trong đầu Vinh lại nảy sinh một câu hỏi khá thú vị: “Xác suất để hàm `binary_search` mà Vinh cài đặt đưa ra kết quả đúng khi mảng đầu vào là chưa được sắp xếp là bao nhiêu?”. Vì xác suất là môn học mà Vinh không thật yêu thích, nên Vinh đã phát biểu lại vấn đề trên dưới dạng bài toán sau đây:

“Có bao nhiêu hoán vị của các số từ 1 đến N mà hàm `binary_search` trả lại đúng chỉ số i sao cho $a[i] = key$?”

Vì số lượng hoán vị như vậy có thể rất lớn nên Vinh chỉ cần biết phần dư của số này khi chia cho 10^9+7 .

Yêu cầu: Hãy giúp Vinh giải quyết bài toán đặt ra.

Dữ liệu: Vào từ thiết bị vào chuẩn gồm một dòng chứa hai số nguyên dương N và key ($key \leq N$).

Kết quả: Ghi ra thiết bị ra chuẩn số nguyên S_N là số lượng hoán vị của các số tự nhiên từ 1 đến N mà hàm `binary_search` cài đặt tìm kiếm nhị phân của Vinh trả lại đúng chỉ số của phần tử có giá trị đúng bằng key . Vì S_N có thể rất lớn nên chỉ cần ghi ra phần dư trong phép chia S_N cho 10^9+7 .

Subtask 1 (0.5 điểm): $N \leq 20$.

Subtask 2 (1.0 điểm): $20 < N \leq 200$.

Subtask 3 (1.5 điểm): $200 < N \leq 30000$.

Subtask 4 (2.0 điểm): $30000 < N \leq 100000$.

Ví dụ:

Dữ liệu	Kết quả
3 1	4

Giải thích: Có sáu hoán vị của ba số 1, 2, 3:

(1, 2, 3), (1, 3, 2), (2, 1, 3), (2, 3, 1), (3, 1, 2), (3, 2, 1).

Hàm `binary_search` của Vinh đối với giá trị $key = 1$ chỉ đưa ra kết quả sai đối với hai hoán vị (2, 3, 1) và (3, 2, 1). Vì vậy câu trả lời là $6 - 2 = 4$.

Bài 6. (5 điểm) Diện tích bao lồi

Cho P là một tập các điểm trên mặt phẳng. Cần phải thực hiện dãy gồm N thao tác trên tập điểm này. Mỗi thao tác thuộc một trong hai dạng sau:

1. Thêm một điểm có tọa độ nguyên vào tập P . Trong thao tác này điểm được thêm vào phải có hoành độ lớn hơn hẳn hoành độ của các điểm có mặt trong tập P tại thời điểm thực hiện thao tác.
2. Xóa khỏi tập P điểm có hoành độ lớn nhất.

Ta gọi bao lồi của tập P gồm một số hữu hạn các điểm trong mặt phẳng là đa giác lồi nhỏ nhất (theo bao hàm) chứa P . Như đã biết, bao lồi của một tập P gồm một số hữu hạn điểm trên mặt phẳng là đa giác lồi với các đỉnh là các điểm thuộc P .

Yêu cầu: Hãy viết chương trình dựa trên dãy N thao tác cần được thực hiện cho trước, sau mỗi thao tác cần tính và đưa ra 2 lần diện tích của đa giác là bao lồi của các điểm trong tập điểm P . Giả thiết rằng: Trước khi thực hiện thao tác đầu tiên tập P là tập rỗng và diện tích của bao lồi của tập rỗng là bằng 0.

Dữ liệu: Vào từ thiết bị vào chuẩn:

- Dòng đầu tiên chứa số nguyên dương N ($N \leq 10^5$) là số lượng thao tác cần thực hiện;
- Dòng thứ i trong số N dòng tiếp theo chứa ba số nguyên T_0 , X_0 và Y_0 ($0 \leq T_0 \leq 1$, $0 \leq X_0, Y_0 \leq 2 \times 10^9$) được ghi cách nhau bởi dấu cách là thông tin cần được giải mã để thực hiện thao

tác thứ i . Các thông số để thực hiện thao tác thứ i được giải mã như sau: Tính $T=(T_0+S)\%2$, trong đó S là bằng 2 lần diện tích của bao lồi của tập điểm đang có P . Ở bước đầu tiên do P là rỗng nên $S = 0$. Để ý rằng S luôn là số nguyên.

1. Nếu $T = 0$, thì thao tác i là thao tác dạng 1, khi đó tọa độ (X, Y) của điểm được thêm vào được tính theo công thức:

$$X = L + ((S + X_0) \% 2\,000\,000\,001),$$

$$Y = ((Y_0 + S) \% 2\,000\,000\,001) - 1\,000\,000\,000.$$

Ở đây, L là hoành độ lớn nhất trong số các hoành độ của các điểm trong tập P hiện tại.

Nếu P là rỗng, thì đặt $L = -10^9$.

2. Nếu $T = 1$, thì cần thực hiện lần lượt K thao tác dạng 2. Số K được tính theo công thức:

$$K = ((X_0 + Y_0 + S) \% Q) + 1,$$

trong đó Q là số lượng điểm trong tập điểm hiện tại.

Dữ liệu đảm bảo là:

- Các tọa độ của mọi điểm trong tập điểm cần xử lý là nằm trong khoảng giữa -10^9 và 10^9 .
- Khi thực hiện thêm vào điểm mới thì hoành độ của nó luôn lớn hơn hoành độ của mọi điểm của tập điểm hiện có.
- Thao tác dạng 2 (loại bỏ điểm) chỉ phải thực hiện khi tập điểm hiện có là khác rỗng.

Kết quả: Ghi ra thiết bị ra chuẩn N dòng: Dòng thứ i là 2 lần diện tích của bao lồi của tập điểm thu được sau khi thực hiện i thao tác đầu tiên. Ghi số 0, trong trường hợp tập điểm là rỗng.

Subtask 1 (0.75 điểm): $N \leq 300$.

Subtask 2 (0.75 điểm): $N \leq 3000$.

Subtask 3 (1.0 điểm): $N \leq 10^5$; số lượng thao tác dạng 2 (sau khi giải mã, có $T=1$) không quá 100.

Subtask 4 (2.5 điểm): không có thêm điều kiện gì ngoài hạn chế $N \leq 10^5$.

Ví dụ:

Dữ liệu	Kết quả
6	0
0 1000000000 1000000000	0
0 1000000 1001000000	3000000000000
0 1000000 999000000	6000000000000
0 1001500 1000001500	3000000000000
1 85072946 2	0
1 61619603 2	

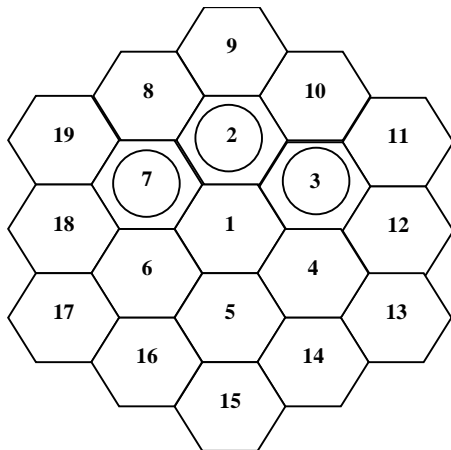
Giải thích: Ta có kết quả giải mã và kết quả thực hiện thao tác sau mỗi một trong 6 bước là:

1. $T = 0, X = 0, Y = 0$. Do sau thao tác thêm điểm này tập $P = \{(0,0)\}$ nên diện tích của bao lồi của nó là 0.
2. $T = 0, X = 10^6, Y = 10^6$. Do sau thao tác thêm điểm này tập $P = \{(0,0), (10^6, 10^6)\}$ nên diện tích của bao lồi của nó là 0.

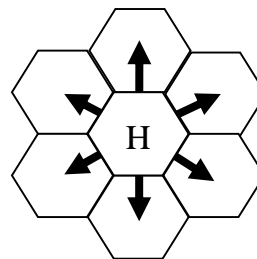
3. $T = 0, X = 2 \cdot 10^6, Y = -10^6$. Do sau thao tác thêm điểm này tập $P = \{(0,0), (10^6, 10^6), (2 \cdot 10^6, -10^6)\}$ nên diện tích của bao lồi của nó (là tam giác với 3 đỉnh tại 3 điểm của P) là $3 \cdot 10^{12}/2$.
4. $T = 0, X = 3000000, Y = 0$. Do sau thao tác thêm điểm này tập $P = \{(0,0), (10^6, 10^6), (2 \cdot 10^6, -10^6), (3 \cdot 10^6, 0)\}$ nên 2 lần diện tích của bao lồi của nó (là tứ giác lồi với 4 đỉnh tại 4 điểm của P) là $6 \cdot 10^{12}$.
5. $T = 1, K = 1$. Do sau khi loại bỏ 1 điểm ta có tập $P = \{(0,0), (10^6, 10^6), (2 \cdot 10^6, -10^6)\}$ nên 2 lần diện tích của bao lồi của nó (là tam giác với 3 đỉnh tại 3 điểm của P) là $3 \cdot 10^{12}$.
6. $T = 1, K = 2$. Do sau khi loại bỏ 2 điểm ta có tập $P = \{(0,0)\}$ nên diện tích của bao lồi của nó là 0.

Bài 7. (5 điểm) Bảo vệ thành phố

Tướng G được giao nhiệm vụ bảo vệ thành phố Alpha có dạng hình lục giác đều. Sau khi khảo sát địa thế và tìm chiến thuật để bảo vệ thành phố, Tướng G chia thành phố thành $3n(n+1)+1$ khu vực có hình dạng ô tổ ong lục giác đều bằng nhau, phân bố đều xung quanh khu vực trung tâm. Các khu vực (các ô tổ ong) được đánh số 1, 2, 3, ..., $3n(n+1)+1$ theo đường xoắn tròn ốc cùng chiều kim đồng hồ, bắt đầu từ khu vực trung tâm (xem minh họa với $n=2$ ở hình 1). Một khu vực có thể có nhiều nhất 6 khu vực chung cạnh lân cận (xem minh họa ở hình 2). Tướng G đã cho đặt m bộ pháo giống hệt nhau ở m khu vực phân biệt có chỉ số p_1, p_2, \dots, p_m ($1 \leq p_1, p_2, \dots, p_m \leq 3n(n+1)+1$).



Hình 1. Bản đồ với $n = 2$ và ba bộ pháo



Hình 2. Khu vực H và 6 khu vực chung cạnh lân cận

Tuy nhiên, sau khi nhận được những tin tức mới nhất từ quân báo về khả năng tấn công của địch, Tướng G quyết định thay đổi vị trí đặt các bộ pháo. Cụ thể, thay vì đặt các bộ pháo ở các khu vực p_1, p_2, \dots, p_m , theo kế hoạch mới, chúng sẽ được đặt ở m khu vực phân biệt có chỉ số q_1, q_2, \dots, q_m ($1 \leq q_1, q_2, \dots, q_m \leq 3n(n+1)+1$). Để thực hiện việc di chuyển các bộ pháo sang các vị trí mới, Tướng G cần điều động các đội vận chuyển. Với mục đích giữ bí mật về các vị trí mới của các bộ pháo, mỗi đội vận chuyển được điều động chỉ thực hiện việc di chuyển bộ pháo từ một khu vực sang khu vực lân cận kề cạnh không có bộ pháo và sau khi hoàn thành nhiệm vụ này đội tự giải thể.

Yêu cầu: Cho biết danh sách ban đầu về các vị trí đặt các bộ pháo và danh sách mới về các vị trí đặt các bộ pháo, hãy giúp Tướng G đưa ra phương án thực hiện việc di chuyển các bộ pháo đến các vị trí mới mà phải điều động ít đội vận chuyển nhất.

Dữ liệu: Vào từ thiết bị vào chuẩn: Dòng thứ nhất chứa số nguyên dương T ($T \leq 5$) là số lượng bộ dữ liệu. Mỗi nhóm dòng trong T nhóm dòng tiếp theo mô tả một bộ dữ liệu theo khuôn dạng sau:

- Dòng đầu chứa số nguyên dương n và m ($m < 3n(n+1)+1$);
- Dòng thứ hai gồm m số nguyên dương p_1, p_2, \dots, p_m ($1 \leq p_1, p_2, \dots, p_m \leq 3n(n+1)+1$);
- Dòng thứ ba gồm m số nguyên dương q_1, q_2, \dots, q_m ($1 \leq q_1, q_2, \dots, q_m \leq 3n(n+1)+1$).

Hai số liên tiếp trên cùng dòng được ghi cách nhau bởi dấu cách.

Kết quả: Ghi ra thiết bị ra chuẩn T nhóm dòng, mỗi nhóm là kết quả tìm được cho bộ dữ liệu tương ứng trong dữ liệu vào, theo khuôn dạng sau:

- Dòng đầu tiên ghi ra số nguyên không âm s là số lượng đội cần điều động để thực hiện việc di chuyển các bộ pháo;
- Tiếp đến là s dòng mô tả nhiệm vụ của s đội. Mỗi dòng ghi 2 số nguyên u, v cách nhau bởi dấu cách cho biết cần thực hiện việc di chuyển bộ pháo từ khu vực u sang khu vực v .

Nếu có nhiều phương án thực hiện thì chỉ cần đưa ra một cách bất kì.

Subtask 1 (1.0 điểm): $n = 2$.

Subtask 2 (1.0 điểm): $n \leq 15$ và $m \leq 15$.

Subtask 3 (2.0 điểm): $n \leq 15$.

Subtask 4 (1.0 điểm): $n \leq 50$.

Ví dụ:

Dữ liệu	Kết quả
1	3
2 3	7 1
7 2 3	1 5
2 1 5	3 1

Giải thích:

Tướng G cần điều động ít nhất 3 đội vận chuyển. Việc di chuyển 3 bộ pháo được đặt ở ba khu vực 7, 2, 3 sang ba vị trí mới được tiến hành như sau:

- Bộ pháo ở khu vực 2 giữ nguyên vị trí;
- Điều động một đội vận chuyển để di chuyển bộ pháo ở khu vực 7 sang khu vực 1. Tiếp đến, điều động một đội vận chuyển khác để di chuyển bộ pháo này sang khu vực 5.
- Cuối cùng, điều động một đội vận chuyển nữa để di chuyển bộ pháo đặt ở khu vực 3 sang khu vực 1.

Tổng cộng, theo phương án này cần điều động 3 đội vận chuyển.

Bài 8. (5 điểm) Trò chơi trên đa giác

Thi đấu với máy tính trong các trò chơi đối kháng luôn tạo ra nhiều cảm hứng cho người chơi. Lần này Hùng gặp phải một trò chơi học búa với máy tính sau đây.

Trò chơi diễn ra trên một đa giác lồi n đỉnh được đánh số từ 1 đến n theo chiều kim đồng hồ, trong đó người ta đã vẽ m đường chéo sao cho không có hai đường chéo nào cắt nhau ở trong đa giác và không có 3 đỉnh nào đôi một có đoạn nối (có thể là cạnh hoặc đường chéo của đa giác).

Hai đấu thủ luân phiên thực hiện nước đi. Đến lượt mình, người chơi phải vẽ thêm một đường chéo của đa giác sao cho đường chéo này không cắt bất cứ đường chéo nào đã vẽ trước đó tại điểm nằm trong đa giác và đồng thời không có bất cứ tam giác nào được tạo ra (nghĩa là không xuất hiện 3 đỉnh nào của đa giác đôi một có đoạn nối). Người đến lượt mà không có cách thực hiện được lượt đi là người thua cuộc và tất nhiên, đối thủ của anh ta là người giành phần thắng.

Yêu cầu: Hùng được quyền lựa chọn là người thực hiện nước đi trước hay sau, hãy giúp Hùng lựa chọn quyền thực hiện nước đi trước hay sau và thực hiện các lượt chơi để giành phần thắng.

Chương trình phải sử dụng một thư viện riêng. Thư viện bao gồm các file sau: **pgamelib.pas** (cho Pascal), **pgamelib.h** (cho C++). Trong chương trình của bạn cần khai báo các thư viện này ở đầu chương trình:

- **uses pgamelib;** đối với Pascal;
- **#include "pgamelib.h"** đối với C++.

Thư viện cung cấp các hàm sau:

- **Các hàm khởi tạo trò chơi**

- **procedure get(var n:longint; var m:longint);** đối với Pascal,
void get(int &n, int &m); đối với C++.

Chương trình phải gọi hàm này để khởi tạo trò chơi. Hàm này trả về một giá trị n là số lượng đỉnh của đa giác và giá trị m là số lượng đường chéo đã nối của đa giác ban đầu. Hàm này chỉ được gọi một lần duy nhất.

- **procedure get_uv(var u:longint; var v:longint);** đối với Pascal,
void get_uv(int &u, int &v); đối với C++.

Chương trình phải gọi hàm này m lần để lấy về chỉ số hai đầu mút của m đường chéo đã vẽ trong đa giác ban đầu.

- **Hàm thực hiện lượt chơi**

procedure play(i:longint; j:longint; var u:longint;
var v:longint); đối với Pascal,
void play(int i, int j, int &u, int &v); đối với C++.

Bạn dùng thủ tục này đưa vào i, j là hai đầu mút của đường chéo được nối trong lượt chơi của mình và thủ tục sẽ trả về u, v là hai đầu mút của đường chéo được nối trong lượt chơi tiếp theo của máy. Nếu bạn lựa chọn chơi trước thì bước đầu tiên gọi hàm **play(i, j, u, v)** với i, j là

nguyên dương. Nếu bạn lựa chọn chơi sau thì bước đầu tiên gọi hàm `play(0,0,u,v)`. Chương trình sẽ kết thúc nếu bạn vi phạm luật chơi hoặc không thể thực hiện tiếp lượt chơi.

Bạn có thể xem các file được cung cấp trên hệ thống để hiểu rõ hơn về cách tương tác với hệ thống.

Subtask 1 (1.0 điểm): $n \leq 20$.

Subtask 2 (0.5 điểm): $n \leq 1000$ và giả thiết thêm rằng: n là số chẵn và không có đường chéo nào trong đa giác xuất phát.

Subtask 3 (1.0 điểm): $n \leq 1000$.

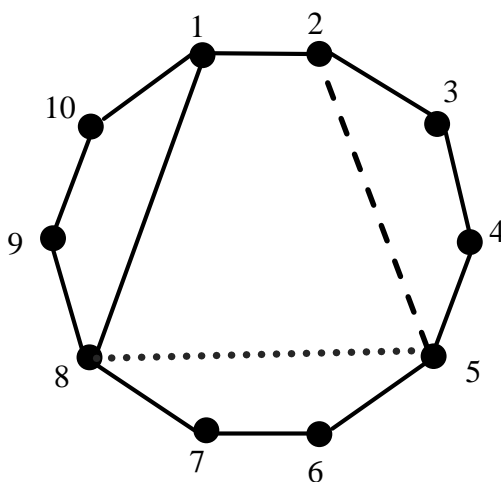
Subtask 4 (1.0 điểm): $n \leq 100000$ và giả thiết thêm rằng: các đường chéo đã vẽ trong đa giác xuất phát chia đa giác ra thành các đa giác con, trong đó mỗi đa giác con đều có số đỉnh nhỏ hơn 8.

Subtask 5 (1.5 điểm): $n \leq 100000$.

Ví dụ:

Gọi hàm	Giá trị trả về
<code>get(n,m)</code>	<code>n=10; m=1;</code>
<code>get_uv(u,v)</code>	<code>u=1; v=8;</code>
<code>play(0,0,u,v)</code>	<code>u=2; v=5;</code>
<code>play(5,8,u,v)</code>	<code>u=0; v=0;</code> Kết thúc chương trình. Bạn giành phần thắng.

Giải thích: Hình 3 dưới đây minh họa cho ví dụ, trong đó các đoạn đen liền nét là thông tin đầu vào. Máy thực hiện nước đi trước và vẽ cạnh (2,5). Đến lượt mình, người chơi vẽ cạnh (5,8). Khi đó các đường chéo (1,8), (2,5) và (5,8) đã vẽ trong đa giác chia đa giác ra thành 4 tứ giác con và người chơi giành phần thắng.



Hình 3. Minh họa cho trò chơi trên đa giác

----- HẾT -----

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