

# Học chung TST2021 buổi 1

Trịnh Hữu Gia Phúc

26/02/2021

Đề bài gồm có 05 trang.

## A1 - Ước lượng 1 - APPROXIMATE\_1

Số liệu thực tế thường không đẹp dễ như trong toán học, vì vậy người ta thường mô phỏng số liệu thực tế bằng các cấu trúc toán học để dễ dàng xử lí.

Một trong các cách mô phỏng đơn giản nhất là sử dụng một đường thẳng, cụ thể như sau:

- Có  $n$  điểm  $(x_i, y_i)$  ( $0 \leq i < n$ )
- Tìm đường thẳng  $y = ax + b$  sao cho *giá trị tối đa* của sai số với mọi  $i$  là bé nhất.
- Sai số với một điểm  $i$  được tính bằng  $|(ax_i + b) - y_i|$

Hãy tìm kết quả tối ưu với một tập điểm cho trước. Kết quả của bạn được cho là chính xác nếu như sai số tương đối hoặc tuyệt đối không vượt quá  $10^{-6}$ . Nói cách khác, nếu như kết quả của bạn là  $P$ , kết quả của giám khảo là  $Q$  thì bạn được cho là đúng nếu như  $\frac{|P-Q|}{\max(1, |Q|)} \leq 10^{-6}$ .

### Dữ liệu vào

- Dòng đầu tiên chứa một số nguyên  $n$  ( $1 \leq n \leq 10^5$ ) là số lượng điểm.
- Mỗi dòng trong số  $n$  dòng tiếp theo chứa hai số nguyên  $x_i, y_i$  là tọa độ của điểm thứ  $i$  trong dữ liệu  $0 \leq |x_i|, |y_i| \leq 10^6$

### Dữ liệu ra

- In ra một số duy nhất là *sai số tối đa* bé nhất có thể đạt được.

### Ví dụ

Input	Output
3 1 1 2 2 3 3	0
4 1 1 1 2 2 1 2 2	0.5

## Giới hạn

- 50% số test có  $n \leq 1000$
- 50% số test còn lại có  $n \leq 10^5$

## A2 - Ước lượng 2 - APPROXIMATE\_2

Số liệu thực tế thường không đẹp đẽ như trong toán học, vì vậy người ta thường mô phỏng số liệu thực tế bằng các cấu trúc toán học để dễ dàng xử lý.

Một trong các cách mô phỏng đơn giản nhất là sử dụng một đường thẳng, cụ thể như sau:

- Có  $n$  điểm  $(x_i, y_i)$  ( $0 \leq i < n$ )
- Tìm đường thẳng  $y = ax + b$  sao cho *tổng* của sai số với mọi  $i$  là bé nhất.
- Sai số với một điểm  $i$  được tính bằng  $|(ax_i + b) - y_i|$

Hãy tìm kết quả tối ưu với một tập điểm cho trước. Kết quả của bạn được cho là chính xác nếu như sai số tương đối hoặc tuyệt đối không vượt quá  $10^{-6}$ . Nói cách khác, nếu như kết quả của bạn là  $P$ , kết quả của giám khảo là  $Q$  thì bạn được cho là đúng nếu như  $\frac{|P-Q|}{\max(1, |Q|)} \leq 10^{-6}$ .

## Dữ liệu vào

- Dòng đầu tiên chứa một số nguyên  $n$  ( $1 \leq n \leq 10^5$ ) là số lượng điểm.
- Mỗi dòng trong số  $n$  dòng tiếp theo chứa hai số nguyên  $x_i, y_i$  là tọa độ của điểm thứ  $i$  trong dữ liệu  $0 \leq |x_i|, |y_i| \leq 10^6$

## Dữ liệu ra

- In ra một số duy nhất là *tổng sai số* bé nhất có thể đạt được.

## Ví dụ

Input	Output
3 1 1 2 2 3 3	0
4 1 1 1 2 2 1 2 2	2

## Giới hạn

- 50% số test có  $n \leq 1000$
- 50% số test còn lại có  $n \leq 10^5$

## B - Miền đất số - NUMBER\_LAND

Bài này chỉ hỗ trợ cho C++, nếu các bạn sử dụng các loại ngôn ngữ phong phú hơn thì vẫn có thể thử suy nghĩ.

Trên một miền đất số có một cánh đồng trồng các số ngẫu nhiên có giá trị từ 0 đến 999999. Người ta muốn khảo sát cánh đồng số này, tuy nhiên, cánh đồng rất lớn nên mọi việc sẽ được xử lý bởi một robot.

Vì phần lớn bộ nhớ của robot cần được sử dụng để thực hiện quá trình di chuyển và khảo sát cánh đồng, chỉ có  $2 \times 10^6$  bit để lưu thông tin thu thập được. Gọi bộ nhớ đó là  $b[i]$  ( $0 \leq i < 2 \times 10^6$ ). Ban đầu,  $b[]$  chỉ chứa toàn bit 0. Hãy lập trình các hàm sau để phục vụ cho quá trình khảo sát:

- `void update(bool b[], int x);`
  - Hàm `update` được gọi khi robot gặp được một số  $x$  mới. Hàm này có thao tác trên  $b$  để cập nhật bộ nhớ của robot.
- `void get(bool b[], int &n, int res[]);`
  - Hàm `get` sẽ được gọi sau khi tất cả các thao tác `update`. Hàm này cần ghi vào  $n$  số lượng số đã khảo sát được và ghi tất cả các số đã khảo sát được vào mảng `res`. Thứ tự các số không nhất thiết phải giống với thứ tự khảo sát ban đầu.

Để nộp bài này, các bạn hãy cài đặt các hàm cần thiết trong cùng một file `.cpp` và nộp file đó. Lưu ý rằng file này không được có hàm `main`. Có thể xem thêm ví dụ ở trong hệ thống nộp bài.

### Giới hạn

Số ngẫu nhiên có nghĩa là các test được sinh ngẫu nhiên. Tất cả các bài nộp đều sẽ được chạy trên một bộ test sinh ngẫu nhiên từ trước đó.

- 50% số test có  $n \leq 10^5$
- 50% số test còn lại có  $n \leq 25 \times 10^4$

## C1 - Trò chơi may rủi 1 - GAME\_1

Người ta thường nói cuộc sống cũng giống như một trò chơi, các lựa chọn của bạn sẽ ảnh hưởng rất lớn đến cuộc sống mà bạn có. Gamindustri gần đây cho ra mắt Raifu, một trò chơi mô phỏng cuộc sống đời thật.

Trò chơi nhìn chung có thể được mô phỏng bằng một đồ thị có hướng không chu trình (DAG). Các nút trên đồ thị mô phỏng các trạng thái của game, như là các trạng thái trong cuộc sống. Các cạnh mô tả các lựa chọn mà người chơi có thể thực hiện, giống như các lựa chọn trong cuộc sống. Thiết kế đảm bảo rằng tất cả các nút có thể đến được từ nút xuất phát và tất cả các nút có thể đi đến được nút kết thúc. Người chơi hoàn tất game nếu như đi được từ nút xuất phát đến nút kết thúc.

Dĩ nhiên, trong cuộc sống, không có lựa chọn nào không có rủi ro, nên mỗi cạnh thứ  $i$  có một xác suất thất bại là  $p_i$  ( $0 \leq p_i \leq 1$ ). Vì game mới cho ra mắt, độ chính xác chưa được cao nên nếu một lựa chọn thất bại, người chơi sẽ phải chơi lại từ đầu.

Một người bắt đầu chơi trò chơi, vì người này không biết gì nhiều về trò chơi nên mỗi khi cần lựa chọn, người đó sẽ chọn ngẫu nhiên trong số các lựa chọn có thể, mỗi lựa chọn sẽ có xác suất được chọn như nhau. Hỏi giá trị kì vọng của số lựa chọn mà người chơi cần chọn cho đến khi hoàn tất game là bao nhiêu.

Bài toán trên có thể được mô phỏng lại theo ngôn ngữ tin học như sau:

- Cho đồ thị  $G$  có  $n$  đỉnh và  $m$  cạnh.

- Các nút đánh số từ 1 đến  $n$ , các cạnh đánh số từ 1 đến  $m$ .
  - Cạnh thứ  $i$  có 3 thông tin là  $u_i, v_i, p_i$ , nối từ đỉnh  $u_i$  đến đỉnh  $v_i$ .
  - Khi đi vào một cạnh  $i$ , người chơi sẽ xuất hiện ở đỉnh 1 với xác suất là  $p_i$ , và đỉnh  $v_i$  với xác suất  $1 - p_i$ .
  - Đồ thị đảm bảo là một DAG sao cho từ 1 đi đến được mọi đỉnh, và từ mọi đỉnh đi được đến  $n$ .
- Ở mỗi đỉnh  $u \neq n$ , người chơi sẽ chọn ngẫu nhiên một trong các cạnh và đi vào cạnh đấy.
  - Tính *giá trị kì vọng* của số lượng lần chọn cạnh.

## Dữ liệu vào

- Dòng đầu tiên chứa hai số nguyên  $n$  và  $m$  ( $1 \leq n, m \leq 10^5$ )
- Dòng thứ  $i$  trong số  $m$  dòng tiếp theo chứa 3 số nguyên  $u_i, v_i, q_i$  ( $1 \leq u_i, v_i \leq n; 0 \leq q_i < 100$ ), mô tả cạnh thứ  $i$ . Giá trị  $p_i$  được tính:  $p_i = \frac{q_i}{100}$ .

## Dữ liệu ra

- In ra một dòng duy nhất là *giá trị kì vọng* của số lần quyết định.
- Kết quả của bạn được cho là chính xác nếu như sai số tương đối hoặc tuyệt đối không vượt quá  $10^{-6}$ . Nói cách khác, nếu như kết quả của bạn là  $P$ , kết quả của giám khảo là  $Q$  thì bạn được cho là đúng nếu như  $\frac{|P-Q|}{\max(1, |Q|)} \leq 10^{-6}$ .

## Ví dụ

Input	Output
4 4 1 2 62 1 3 5 2 4 66 3 4 1	3.1130223427129105

## C2 - Trò chơi may rủi 2 - GAME\_2

Người ta thường nói cuộc sống cũng giống như một trò chơi, các lựa chọn của bạn sẽ ảnh hưởng rất lớn đến cuộc sống mà bạn có. Gamindustri gần đây cho ra mắt Raifu, một trò chơi mô phỏng cuộc sống đời thật.

Trò chơi nhìn chung có thể được mô phỏng bằng một đồ thị có hướng không chu trình (DAG). Các nút trên đồ thị mô phỏng các trạng thái của game, như là các trạng thái trong cuộc sống. Các cạnh mô tả các lựa chọn mà người chơi có thể thực hiện, giống như các lựa chọn trong cuộc sống. Thiết kế đảm bảo rằng tất cả các nút có thể đến được từ nút xuất phát và tất cả các nút có thể đi đến được nút kết thúc. Người chơi hoàn tất game nếu như đi được từ nút xuất phát đến nút kết thúc.

Dĩ nhiên, trong cuộc sống, không có lựa chọn nào không có rủi ro, nên mỗi cạnh thứ  $i$  có một xác suất thất bại là  $p_i$  ( $0 \leq p_i \leq 1$ ). Vì game mới cho ra mắt, độ chính xác chưa được cao nên nếu một lựa chọn thất bại, người chơi sẽ phải *chơi lại từ đầu*.

Một người bắt đầu chơi trò chơi, vì người này đã chơi trò chơi nhiều lần nên mỗi khi cần lựa chọn, người đó sẽ chọn lựa chọn sao trung bình cho số lượng lượt cần chọn là ít nhất. Hỏi *giá trị kì*

vọng của số lựa chọn mà người chơi cần chọn cho đến khi hoàn tất game là bao nhiêu nếu chơi tối ưu.

Bài toán trên có thể được mô phỏng lại theo ngôn ngữ tin học như sau:

- Cho đồ thị  $G$  có  $n$  đỉnh và  $m$  cạnh.
  - Các nút đánh số từ 1 đến  $n$ , các cạnh đánh số từ 1 đến  $m$ .
  - Cạnh thứ  $i$  có 3 thông tin là  $u_i, v_i, p_i$ , nối từ đỉnh  $u_i$  đến đỉnh  $v_i$ .
  - Khi đi vào một cạnh  $i$ , người chơi sẽ xuất hiện ở đỉnh 1 với xác suất là  $p_i$ , và đỉnh  $v_i$  với xác suất  $1 - p_i$ .
  - Đồ thị đảm bảo là một DAG sao cho từ 1 đi đến được mọi đỉnh, và từ mọi đỉnh đi được đến  $n$ .
- Ở mỗi đỉnh  $u \neq n$ , người chọn một cạnh sao cho giá trị kì vọng của số lượt sau nước đi này là tốt nhất. Nói cách khác, người này muốn làm giá trị kì vọng số nước đi là bé nhất có thể.
- Tìm giá trị kì vọng số nước đi bé nhất có thể nếu người chơi thực hiện chơi tối ưu.

## Dữ liệu vào

- Dòng đầu tiên chứa hai số nguyên  $n$  và  $m$  ( $1 \leq n, m \leq 10^5$ )
- Dòng thứ  $i$  trong số  $m$  dòng tiếp theo chứa 3 số nguyên  $u_i, v_i, q_i$  ( $1 \leq u_i, v_i \leq n; 0 \leq q_i < 100$ ), mô tả cạnh thứ  $i$ . Giá trị  $p_i$  được tính:  $p_i = \frac{q_i}{100}$ .

## Dữ liệu ra

- In ra một dòng duy nhất là giá trị kì vọng của số lần quyết định nếu người chơi chơi tối ưu (dùng ít nước đi nhất có thể).
- Kết quả của bạn được cho là chính xác nếu như sai số tương đối hoặc tuyệt đối không vượt quá  $10^{-6}$ . Nói cách khác, nếu như kết quả của bạn là  $P$ , kết quả của giám khảo là  $Q$  thì bạn được cho là đúng nếu như  $\frac{|P-Q|}{\max(1, |Q|)} \leq 10^{-6}$ .

## Ví dụ

Input	Output
4 4 1 2 62 1 3 5 2 4 66 3 4 1	2.0733652312599680