

Đoán màu bi

An và Nam đang chơi với nhau một trò chơi.

An có N viên bi được đánh số từ 0 tới $N - 1$, mỗi viên bi có một màu, các màu được ký hiệu bởi các số nguyên dương.

Nhiệm vụ của Nam là xác định được màu của từng viên bi bằng cách hỏi An dạng câu hỏi sau nhiều lần:

- Nam chọn ra một tập các viên bi, sau đó An sẽ nói cho Nam biết số lượng màu phân biệt có trong tập bi đó.

Bạn hãy vào vai Nam tìm một chiến lược chơi để thắng được An.

Cài đặt:

Hãy viết một chương trình (đóng vai Nam) tương tác với chương trình chấm (đóng vai An) trong đó cài đặt hàm:

void play(int N)

- N ($1 \leq N \leq 150$) là số lượng viên bi mà Nam có
- Chương trình chấm sẽ gọi hàm này đúng một lần để bắt đầu trò chơi

Hàm *play* có thể gọi hai hàm sau:

int count_distinct(vector<int> v)

- vector v* sẽ chứa chỉ số các viên bi mà Nam chọn. *v* không được phép chứa hai chỉ số giống nhau, cũng như không được phép chứa các chỉ số không hợp lệ.
- Hàm này sẽ trả về số lượng màu phân biệt trong số các viên bi trong *vector v*.
- Độ phức tạp của hàm này tuyến tính với số lượng phần tử trong *vector v*.

void answer(vector<int> v)

- vector<int> v* chứa đúng N phần tử nguyên dương, trong đó $v[i]$ ($0 \leq i < N$) là màu của viên bi thứ i mà Nam đoán.
- Gọi M là số lượng màu phân biệt của N viên bi, $v[i]$ chỉ nhận các giá trị từ 1 tới M .
- Hàm này chỉ nên được gọi duy nhất một lần và chương trình của bạn nên kết thúc ngay sau đó.

Để gọi được hai hàm trên, chương trình của bạn phải khai báo thư viện *balls.h*

`#include "balls.h"`

Lưu ý chương trình của bạn không viết hàm *main()*.

Ví dụ $N = 6$, chương trình của bạn có thể tương tác với chương trình chấm như sau:

	play(6)
<i>count_distinct([1, 5, 6])</i>	3
<i>count_distinct([1, 3])</i>	1
<i>count_distinct([5, 4])</i>	1
<i>count_distinct([2, 6])</i>	2

<code>count_distinct([4, 2])</code>	1
<code>answer([1, 2, 1, 2, 2, 3])</code>	

Trong ví dụ trên, nếu gọi `answer([2, 3, 2, 3, 3, 1])` thì cũng được tính là đúng.

Chấm điểm:

Gọi Q là số lần gọi hàm `count_distinct` của bạn. Gọi p là điểm của test đang chấm. Một trong các trường hợp sau sẽ xảy ra:

- 1) $Q \leq 1500$: bạn nhận được p điểm
- 2) $1500 < Q \leq 12000$: bạn nhận được $p * 1500 / Q$ điểm
- 3) $Q > 12000$: bạn nhận được 0 điểm.

Chương trình chấm mẫu:

Bạn được cung cấp chương trình chấm mẫu `grader_sample.cpp`, file `balls.h` và một file bài làm mẫu `balls_sample.cpp`.

Chương trình chấm mẫu `grader_sample.cpp` đọc dữ liệu đầu vào theo định dạng:

- Dòng đầu: N
- Dòng hai: N số nguyên là màu của N viên bi.

Nếu chương trình của bạn trả lời không chính xác, trình chấm mẫu sẽ in ra lỗi của bạn.