|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **TT** | **Tên bài** | **File****Chương trình** | **File****dữ liệu** | **File****kết quả** | **Điểm** |
| **1** | **Chọn bi** | **SGAME.\*** | **SGAME.INP** | **SGAME.OUT** | **6,0** |
| **2** | **Chọn đỉnh** | **SADJ.\*** | **SADJ.INP** | **SADJ.OUT** | **7,0** |
| **3**  | **Phân hoạch đẹp** | **PARTITION.\*** | **PARTITION.INP** | **PARTITION.OUT** | **7,0** |

*Phần mở rộng của File chương trình là PAS hoặc CPP tùy theo ngôn ngữ lập trình sử dụng là Pascal hoặc C++*

Cấu hình dịch:

**G++ 4.9.2: -std=c++11 -O2 -s -static -Wl,--stack,66060288 -lm -x c++**

**FPC 3.0.4: -O2 -XS -Sg -Cs66060288**

***Viết chương trình giải các bài toán sau:***

# Bài 1: Chọn bi

An và Bình chơi một trò chơi như sau: Đầu tiên có một dãy gồm $n$ viên bi xếp thành một hàng, mỗi viên bi có một trong hai màu: đỏ hoặc xanh. Hai người chơi luân phiên, An được đi trước.

Mỗi lần người đến lượt chơi sẽ chọn một viên bi ở một trong hai đầu dãy các viên bi còn lại. Trờ chơi sẽ kết thúc nếu như có một người có được $k$ viên bi màu đỏ, khi đó người này sẽ thua.

**Yêu cầu:** Viết chương trình kiểm tra xem An (người được đi trước) có tồn tại phương án chọn bi sao cho luôn thắng hay không?

**Dữ liệu:** Vào từ file văn bản SGAME.INP

Dòng đầu tiên chứa số nguyên dương $T (1\leq T\leq 5)$ là số bộ dữ liệu. Tiếp theo là $T$ nhóm dòng, mỗi nhóm mô tả một bộ dữ liệu với cấu trúc như sau:

* Dòng đầu tiên chứa hai số nguyên dương $n$ và $k $ ($1\leq k<n\leq 350)$
* Dòng thứ hai chứa dãy gồm $n$ ký tự **R** hoặc **B**. Ở đây ký tự **R** ứng với viên bi màu đỏ còn ký tự **B** ứng với viên bi màu xanh.

**Kết quả:** Ghi ra file văn bản SGAME.OUT

Gồm $T$ dòng, dòng thứ $i$ ghi "YES" (không có dấu nháy kép) nếu An có phương án thắng trong bộ dữ liệu thứ $i$, ngược lại ghi "NO" (không dấu nháy kép). Các bộ dữ liệu đánh số theo thứ tự xuất hiện trong input.

**Ràng buộc:**

* Có 20% số test ứng với 20% số điểm của bài có $1\leq n\leq 20$
* 30% số test tiếp theo ứng với 30% số điểm của bài có $1\leq n\leq 50$
* 50% số test còn lại không có ràng buộc bổ sung.

# Bài 2. Chọn đỉnh

Cho đồ thị vô hướng liên thông $n$ đỉnh (đánh số 1, 2, ..., $n$) và có đúng $n$ cạnh. Không có cạnh nào nối một đỉnh với chính nó và giữa hai đỉnh bất kỳ có tối đa một cạnh nối trực tiếp.

Một tập $S$ các đỉnh của đồ thị được gọi là một tập đẹp nếu như với mọi đỉnh của đồ thị đều kề với chính xác một đỉnh của tập $S$

**Yêu cầu:** Hãy tìm tập đẹp $S$ có lực lượng nhỏ nhất hoặc chỉ ra không tồn tại một tập như vậy.

**Dữ liệu:** Nhập vào từ file văn bản SADJ.INP

* Dòng đầu tiên chứa số nguyên dương $n$
* Trong $n$ dòng tiếp theo mỗi dòng chứa hai số nguyên là số hiệu hai đỉnh đầu mút một cạnh. Mỗi cạnh được mô tả đúng một lần trong dữ liệu vào.

**Kết quả:** Ghi ra file văn bản SADJ.OUT

Nếu không tồn tại tập đẹp, ghi -1. Ngược lại, ghi một số nguyên là lực lượng nhỏ nhất của một tập đẹp.

**Ràng buộc:**

* Có 10% số test của bài ứng với 10% số điểm mỗi đỉnh kề với đúng hai đỉnh khác
* 20% số test tiếp theo ứng với 10% số điểm của bài có $3\leq n\leq 20$
* 30% số test tiếp theo ứng với 35% số điểm của bài có $3\leq n\leq 1000$
* 40% số test còn lại có $3\leq n\leq 10^{5}$

*Ví dụ:*

|  |  |
| --- | --- |
| **SADJ.INP** | **SADJ.OUT** |
| **4****1 2****2 3****3 4****4 1** | **2** |
| **3****1 2****2 3****3 1** | **-1** |
| **7****1 2****2 3****3 4****4 5****5 6****6 7****2 4** | **4** |

# Bài 3. Phân hoạch đẹp

Trên trục số cho $n$ đoạn thẳng $\left[a\_{1}, b\_{1}\right], \left[a\_{2}, b\_{2}\right],…,[a\_{n}, b\_{n}]$. Một phân hoạch là một cách chia các đoạn thẳng thành các nhóm, mỗi đoạn thẳng nằm trong đúng một nhóm, mỗi nhóm gồm các đoạn thẳng liên tiếp sao cho không có hai đoạn thẳng nào trong chúng giao nhau. Một phân hoạch đẹp là phân hoạch có số nhóm là ít nhất

Viết chương trình thực hiện $Q$ truy vấn, mỗi truy vấn được mô tả bằng hai số nguyên $u, v (1\leq u\leq v\leq n)$ thể hiện yêu cầu:  **Hãy ính số nhóm trong phân hoạch đẹp của dãy các đoạn thẳng** $\left[a\_{u}, b\_{u}\right], \left[a\_{u+1}, b\_{u+1}\right],…,[a\_{v}, b\_{v}]$

**Dữ liệu:** Nhập vào từ file văn bản PARTITION.INP

* Dòng đầu tiên chứa số nguyên dương $n$
* $n$ dòng tiếp theo, dòng thứ $i$ chứa hai số nguyên $a\_{i}, b\_{i}$ $\left(1\leq a\_{i}\leq b\_{i}\leq 10^{9}\right)$
* Dòng tiếp theo chứa số nguyên dương $Q$ - số truy vấn
* $Q$ dòng cuối cùng, dòng thứ $j$ chứa hai số nguyên dương $u\_{j}, v\_{j} (1\leq u\_{i}\leq v\_{j}\leq n)$ mô tả truy vấn thứ $j$ ($j=1÷n) $

**Kết quả:** Ghi ra file văn bản PARTITION.OUT

Gồm $Q$ dòng, dòng thứ $i$ ghi một số nguyên là kết quả tìm được cho truy vấn thứ $i (i=1÷n)$

**Ràng buộc:**

Trong tất cả các test : $1\leq n, Q\leq 2.10^{5}$

* Có 10% số test ứng với 10% số điểm của bài có $Q=1, u\_{1}=1, v\_{1}=n$
* 10% số test tiếp theo ứng với 10% số điểm của bài có $1\leq n,Q\leq 5000$
* 20% số test tiếp theo ứng với 20% số điểm của bài có $1\leq n\leq 5000, 1\leq Q\leq 2.10^{5}$
* 20% số test tiếp theo ứng với 20% số điểm của bài có $1\leq n\leq 2.10^{5}, 1\leq Q\leq 5000$
* 40% số test còn lại không có ràng buộc bổ sung

*Ví dụ:*

|  |  |
| --- | --- |
| **PARTITION.INP** | **PARTITION.OUT** |
| **5****1 3****3 3****4 6****2 3****1 1****3****1 4****3 5****1 5** | **3****1****3** |

---HẾT---