**KHOÁ HỌC: THUẬT TOÁN CHUYÊN SÂU – SAMSUNG PTIT – HÈ 2025**

**Buổi 12. Đề Ôn tập số 4**

\*\*\*

Thứ 7, ngày 06 tháng 09 năm 2025

**Giảng viên:** TS. Đỗ Đức Đông – Trường Đại học Công nghệ - Đại học Quốc gia Hà Nội

\*\*\*

**ĐỀ BÀI**

**BÀI 1. D5REC**

Cho n hình chữ nhật, hình thứ i có kích thước ai x bi. Hình chữ nhật kích thước a x b có thể lồng vào hình chữ nhật kích thước a’ x b’ nếu (a ≤ a’ và b ≤ b’) hoặc (a ≤ b’ và b ≤ a’). Nhiệm vụ của bạn là hãy tìm cách chọn ra 1 tập gồm nhiều hình chữ nhật nhất lần lượt gồm các hình xếp lồng vào nhau, mỗi hình chữ nhật chỉ chứa trực tiếp một hình chữ nhật khác.

**Input**

* Dòng đầu chứa số nguyên n là số hình chữ nhật (n ≤ 105)
* n dòng tiếp theo, mỗi dòng gồm 2 số nguyên ai và bi thể hiện kích thước của hình chữ nhật thứ i. (1 ≤ i ≤ n, 1 ≤ ai, bi ≤ 109)

**Output**

* Dòng đầu tiên ghi ra số k là số hình chữ nhật chọn được
* Dòng thứ hai ghi ra lần lượt k số, mỗi số cách nhau một khoảng trống, các số theo số thứ tự hình từ nhỏ đến lớn

**Ví dụ**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 4  2 3  1 2  1 4  3 3 | 3  2 1 4 |

**Giải thích test:**

|  |  |
| --- | --- |
|  | Phương án xếp tối ưu là: Hình 4 dưới cùng, đè hình 1 lên hình 4, đè hình 2 lên hình 1 |

**BÀI 2. DÃY CHỮ SỐ**

An và Bình cùng chơi trò chơi với dãy số như sau: An viết liên tiếp một dãy gồm n chữ số thập phân, tiếp theo Bình tách dãy chữ số trên thành các nhóm chữ số để nhận được một dãy số. Sau đó cả hai bạn cùng tiến hành tìm dãy con tăng dài nhất từ dãy số mới nhận được.

Ví dụ: An viết dãy chữ số thập phân 314159265358979:

* Nếu Bình tách dãy trên thành dãy số gồm 6 số: 3, 14, 159, 26, 53, 58979 thì cả hai bạn sẽ tìm được dãy con tăng dài nhất gồm 5 số là: 3, 14, 26, 53, 58979.
* Nhưng nếu Bình tách thành dãy số gồm 10 số: 3, 1, 4, 1, 5, 9, 26, 53, 58, 979 thì cả hai bạn sẽ tìm được dãy con tăng dài nhất gồm 8 số là: 3, 4, 5, 9, 26, 53, 58, 979.

Nhiệm vụ của bạn là hãy giúp Bình tìm cách tách dãy số của An sao cho dãy con tăng dài nhất của dãy số sau khi tách gồm nhiều số nhất có thể

**Input**

* Dòng đầu chứa số nguyên dương n là số chữ số thập phân của dãy s (n ≤ 103)
* Dòng thứ hai là n chữ số thập phân s[i], các số được viết liền nhau (1 ≤ s[i] ≤ 9)

**Output**. Một số nguyên duy nhất là số phần tử nhiều nhất của dãy con tăng dài nhất

**Ví dụ**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 15  314159265358979 | 8 |
| 10  123456789 | 9 |

**Giải thích test**

* **Test 1:** Đã giải thích ở đề bài;
* **Test 2:** Tách dãy số thành: 1, 2, 3, 4, 5, 6, 7, 8, 9. Cả dãy số đều tăng.

**BÀI 3. VECTORG**

Cho n véc tơ, mỗi véc tơ có m thành phần. Hai véc tơ U (u1, u2, …, um) và véc tơ V (v1, v2, …, vm) được gọi là có “quan hệ” với nhau nếu tồn tại 2 véc tơ A và B trong đó:

* A (a1, a2, …, am) là hoán vị của m thành phần trong véctơ U
* B (b1, b2, …, bm) là hoán vị của m thành phần trong véctơ V
* Với mọi i mà 1 ≤ i ≤ m thì ai ≤ bi. Kí hiệu là A ≤ B

Nhiệm vụ của bạn là hãy chia n véc tơ này thành ít nhóm nhất có thể sao cho trong mỗi nhóm thì 2 véc tơ bất kỳ k có quan hệ với nhau

**Input**

* Dòng đầu ghi 2 số n và m là số véc tơ và số lượng thành phần của mỗi véc tơ
* n dòng tiếp theo, mỗi dòng gồm m số biểu thị 1 véc tơ (n ≤ 1000, m ≤ 20)

**Output.**

* Dòng đầu tiên ghi số nhóm ít nhất được chia k
* k dòng tiếp theo, mỗi dòng là một nhóm, gồm x + 1 số có dạng như sau:
* Số đầu tiên của nhóm chính là số phần tử của nhóm
* x số tiếp theo là số thứ tự của véc tơ được chọn vào nhóm đó theo thứ tự tăng dần
* Thí sinh được phép liệt kê các nhóm theo bất kỳ thứ tự nào

**Ví dụ.**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 4 3  1 1 1  1 2 3  2 2 2  2 1 1 | 3  1 1  2 2 3  1 4 |

**Giải thích test**

* Véc tơ 1 (V1) và V2: (1, 1, 1) ≤ (1, 2, 3) 🡪 V1 và V2 khác nhóm
* V1 và V3: (1, 1, 1) ≤ (2, 2, 2) 🡪 V1 và V3 khác nhóm
* V1 và V4: Sắp xếp V4 lại thành (1, 1, 2) thì (1, 1, 1) ≤ (1, 1, 2) 🡪 V1 và V4 khác nhóm

Vậy V1 phải đứng riêng so với V2, V3 và V4. Có {V1}, {V2, V3, V4}

* V2 và V3: (1, 2, 3) và (2, 2, 2). Không có hoán vị nào của V2 thỏa mãn V2 ≤ V3 hoặc ngược lại 🡪 V2, V3 chung nhóm
* V2 và V4. Sắp xếp lại V4 thành (1, 1, 2) thì (1, 2, 3 ) ≥(1, 1, 2) **🡪** V2, V4 khác nhóm

Có {V1}, {V2. V3}, {V4}

* V3 và V4: Hiển nhiên (2, 2, 2) ≥ (2, 1, 1) 🡪 V3 khác nhóm V4

Vậy {V1}, {V2. V3}, {V4} là cách chia tối ưu

**BÀI 4. DPAIR**

Cho dãy gồm số n + 1 (n ≤ 100) nguyên dương A = (a0, a1, …, an), các phần tử có giá trị không vượt quá 2n và một biến d ban đầu nhận giá trị bằng 1.

Viết hàm **void solve(int n){}** có **#include "dpair.h"** và sử dụng các hàm sau để tìm một cặp phần tử chia hết cho nhau.

1. **void msetd(int v):** Gán d = v ∈ [1, 2n]
2. **bool mcheck(int i):** Kiểm tra ai có chia hết cho d không
3. **void mseti(int i):**

* Nếu ai chia hết cho d thì gán ai = ai / d
* Ngược lại, giữ nguyên ai

1. **bool mcheckdup(string s):**

Kiểm tra xem các phần tử thuộc tập s trên dãy có phần tử nào bằng nhau không?

Tập s mô tả bằng một xâu nhị phân độ dài n + 1, kí tự thứ i bằng ‘1’ nếu ai thuộc tập, ngược lại bằng ‘0’

1. **void mtell(int i, int j):** Trả lời cặp phần tử i, j giá trị ban đầu chia hết cho nhau.

**Tính điểm.**

* Các hàm sẽ được thử nghiệm nhiều lần.
* Các hàm 1, 2, 3, 4 mỗi hàm có chi phí bằng 1, cần tối ưu chi phí.
* Hàm 5 chỉ gọi đúng một lần để trả lời cho một lần thử nghiệm.

**BÀI 5. OLIUCAKE**

Một chiếc bánh hình chữ nhật kích thước m x n ô. Các hàng được đánh số từ 1 đến m, các cột được đánh số từ 1 đến n. Ô nằm trên hàng i (1 ≤ i ≤ m), cột j (1 ≤ j ≤ n) là một mẩu bánh loại socola a(i, j). Biết rằng mỗi loại socola xuất hiện đúng 4 lần.

Oliu muốn cắt chiếc bánh thành các miếng bánh, mỗi miếng bánh có đúng hai mẩu bánh là hai ô kề cạnh. Sau khi cắt thành các miếng bánh, Oliu đã tìm ra một cách chia các miếng bánh thành hai phần mà ở mỗi phần mỗi loại socola xuất hiện đúng hai lần. Bài toán không hề đơn giản, Oliu đố bạn tìm được cách chia thỏa mãn

**Input**

* Dòng đầu tiên gồm 2 số m và n (m, n ≤ 1000)
* m dòng tiếp theo, mỗi dòng gồm n số lần lượt là trị số của loại sô cô la ở hàng (i, j)

**Output**

Gồm dòng mô tả phần thứ nhất, mỗi dòng chứa bốn số x, y, u, v là hai ô kề nhau mô tả miếng bánh gồm hai mẩu bánh.

**Ví dụ**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 2 4  **1** 2 1 **2**  **2** 2 1 **1** | 1 1 2 1  1 4 2 4 |

**//1166000+150000 + 90000 (Dịch vụ)=1390000**