

Very Simple OCR System

Hiện tại, bài tập này đã có trên online judge chính thức của VNOI, bạn có thể truy cập ở đây: <https://oj.vnoi.info/problem/vmvsos>

Trong bài toán này chúng ta làm quen với Optical Character Recognition (OCR – nhận dạng chữ trong ảnh) và tìm cách thiết lập một hệ thống OCR đơn giản.

Có một bức ảnh hình chữ nhật được số hóa bằng một ma trận có W cột và H dòng. Trong bức ảnh ghi một thông điệp. Hệ thống OCR cần phải tìm ra thông điệp này.

Để đơn giản hóa bài toán, các quy tắc sau được đề ra:

- Bức ảnh chỉ gồm hai màu đen và trắng. Màu trắng là màu nền. Thông điệp được viết bằng màu đen. Mọi điểm đen của bức ảnh đều thuộc thông điệp.
- Thông điệp chỉ gồm các chữ cái tiếng Anh thường. Không có dấu cách (space).
- Thông điệp được viết bằng máy tính nên rất đều và ngay ngắn: viết trên một dòng duy nhất; các chữ cái không đè lên nhau, cũng không chạm vào nhau.
- Người viết thông điệp này đã sử dụng lẫn lộn mười một loại font khác nhau, bao gồm: Arial, Book Anquita, Bookman Oldstyle, Calibri, Comic San MS, Curlz MT, Gentium Basic, Impact, Monotype Corsiva, Papyrus, Times New Roman. Tuy nhiên, anh ta chỉ dùng đúng một font-size là cỡ 28.
- Bức ảnh có nhiễu, nghĩa là có một số điểm (trong thông điệp gốc) bị chuyển từ đen thành trắng. Số điểm bị đổi không quá 10% của bức ảnh.

Input

- Dòng 1 chứa hai số W và H ($W \times H < 65,536$).
- H dòng sau mỗi dòng chứa W chữ cái mô tả bức ảnh. Dấu chấm (.) thể hiện điểm trắng. Dấu thăng (#) thể hiện điểm đen.
- Bạn có thể download mẫu của 11 font trên tại: [link](#)

Output

- Một dòng duy nhất là thông điệp tìm được.

Giới hạn

- Trong 30% test, không có bức ảnh nào bị nhiễu (không có điểm nào bị chuyển từ đen thành trắng).
- **Giới hạn mã nguồn: 16KB.**

Example

Input 1:

66 22

```
.....
...#####.....
...#####.....
...########.....
...#####.....#####.....
...#####.....#####.....#####.....
...#####.....#####.....###.###.##.###.....
...#####.....###.....#####.....#####.....#####.....
...#####.....###.....#####.....#####.....
...#####.....#####.....#####.....
...#####.....###.....#####.....#####.....
...#####.....#####.....#####.....#####.....
...#####.....#####.....#####.....#####.....
...#####.....#####.....#####.....#####.....
...#####.....#####.....#####.....#####.....
...#####.....#####.....#####.....#####.....
...#####.....#####.....#####.....#####.....
...#####.....#####.....#####.....#####.....
...#####.....#####.....#####.....#####.....
...#####.....#####.....#####.....#####.....
...#####.....#####.....#####.....#####.....
...#####.....#####.....#####.....#####.....
.....
```

Output 1:

aaa

Input 2:

56 27

```
.....
...####.####.....
...#####.....
...#####.###.....
...#####.....
...#####.#####.....
...#####.#####.....#.#####.....
...#####.#####.....#.#####.....#####.....
...#####.#####.....#####.###.###.....
...#####.#####.....#####.###.###.....#####.....
...#####.###.#####.#####.....#.#####.....#####.....
...#####.#####.....#####.#####.#####.....
...#####.###.#####.....#.###.#####.#####.....
...#####.#####.....#####.###.###.....#####.....
...#####.#####.###.....###.#####.....#####.....
...#####.#####.....###.....#####.....
...#####.#####.....###.....#####.....
...###.###.###.###.....###.....#####.....###.
...#####.#####.....#.#####.....#####.....
...#####.#####.....###.....#####.....#####.....
...#####.#####.....#.#####.....#####.....
...#####.#####.....###.....#####.###.....
...#####.#####.....###.###.....#####.....
...#####.....#####.....#####.....
.....
.....#####.....
.....
```

Output 2:

orc