




Mini Tetris

Như bạn đã biết, trò chơi máy tính phổ biến "Tetris" được phát minh bởi lập trình viên Nga Alexey Pajitnov. Trong bài toán này, bạn cần phải viết chương trình chơi phiên bản đơn giản của trò chơi này.

Sân chơi là một hình chữ nhật đặt dọc, được gọi là "giếng". Các hình ngẫu nhiên gồm các khối vuông đơn vị xuất hiện trên đỉnh của giếng, người chơi chọn vị trí nằm ngang và quay hình, sau đó hình rơi xuống giếng. Mục tiêu của trò chơi là tạo ra các đường ngang không có khoảng trống. Khi một đường ngang như vậy được tạo ra, nó sẽ biến mất, và bất kỳ khối nào trên đường ngang bị xóa sẽ rơi xuống.

Trong phiên bản thay đổi của trò chơi này, kích thước giếng là 3×4 đơn vị, và chỉ có ba loại hình:

Loại	Hình
1	
2	
3	

Bạn sẽ thua cuộc nếu tại một thời điểm có năm đường ngang không trống của giếng. Bạn giành chiến thắng nếu bạn không bị thua cuộc sau n hình đã rơi xuống.

Bạn cần phải viết chương trình chơi trò chơi mô tả ở trên và thắng bất kể hình nào sẽ xuất hiện.

Implementation details

Bạn cần xây dựng bốn hàm (thủ tục):

- `void init(int n)`. Hàm này được gọi trước khi bất kỳ hàm nào khác.
- `void new_figure(int figure_type)`. Hàm này được gọi khi hình mới xuất hiện. `figure_type` là một số **1** đến **3**, chỉ ra loại hình từ bảng trên.
- `int getPosition()`. Hàm này nên trả về một số từ **0** đến **2**, là vị trí khối cuối cùng trái nhất của hình.

- `int getRotation()`. Hàm này nên trả một số từ **0** đến **3**, là số lần quay hình ngược chiều kim đồng hồ.

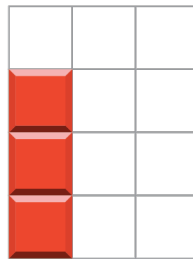
Hàm `getPosition` và `getRotation` sẽ chỉ được gọi sau `new_figure`.

Hãy sử dụng các file mẫu được cung cấp để biết chi tiết thực hiện trong ngôn ngữ lập trình của bạn.

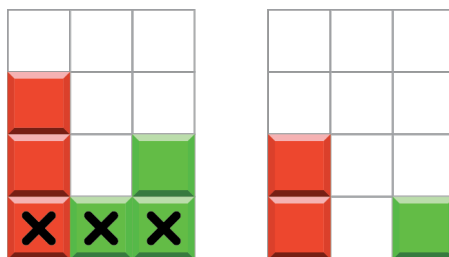
Example

grader tạo các lời gọi hàm sau:

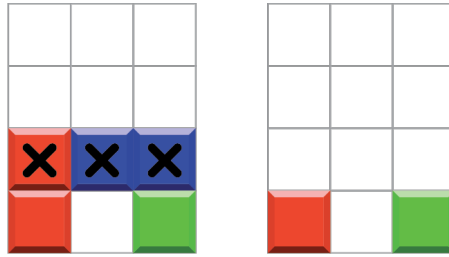
- `init(3)`. Sẽ có ba hình.
- `new_figure(1)`. Hình loại **1** sẽ rơi từ trên đỉnh của giếng.
- `getPosition()` trả về **0**. Nghĩa là người chơi muốn đặt hình ở cột trái nhất;
- `getRotation()` trả về **1** (hoặc **3**). Nghĩa là người chơi muốn quay hình thẳng đứng.
- Sau đó hình rơi xuống, giếng có dạng như sau.



- `new_figure(2)`.
- `getPosition()` trả về **1**.
- `getRotation()` trả về **1**.
- Sau khi hình rơi xuống, dòng đầu tiên sẽ đầy, do đó nó biến mất và giếng có dạng như sau.



- `new_figure(1)`.
- `getPosition()` trả về **1**.
- `getRotation()` trả về **0** (hoặc **2**).
- Sau khi hình rơi xuống, dòng thứ hai sẽ đầy, do đó nó biến mất và giếng có dạng như sau.



Subtasks

Trong tất cả các subtask $n \leq 1000$.

1. (7 points) Tất cả các hình đều là loại 1 ,
2. (13 points) Tất cả các hình đều là loại 2 ,
3. (21 points) Tất cả các hình đều là loại 3 ,
4. (53 points) Các hình có thể khác loại.

Sample Grader

Chương trình mẫu grader đọc dữ liệu đầu vào theo định dạng sau:

- dòng 1: Một số nguyên n .
- dòng 2: n số nguyên: là số mô tả loại hình.

Language Notes

Hãy sử dụng các file mẫu được cung cấp để biết chi tiết thực hiện trong ngôn ngữ lập trình của bạn.