

Contributor : SHANG-YAN XUN

Subject : Shisensho

Main testing concept :

Basics

- C++ BASICS
- FLOW OF CONTROL
- FUNCTION BASICS
- PARAMETERS AND OVERLOADING
- ARRAYS
- STRUCTURES AND CLASSES
- CONSTRUCTORS AND OTHER TOOLS
- OPERATOR OVERLOADING, FRIENDS, AND REFERENCES
- STRINGS
- POINTERS AND DYNAMIC ARRAYS

Functions

- SEPARATE COMPILATION AND NAMESPACES
- STREAMS AND FILE I/O
- RECURSION
- INHERITANCE
- POLYMORPHISM AND VIRTUAL FUNCTIONS
- TEMPLATES
- LINKED DATA STRUCTURES
- EXCEPTION HANDLING
- STANDARD TEMPLATE LIBRARY
- PATTERNS AND UML

Description :

Shisensho(四川省) is a classic tile-matching game. The objective of the game is to match all the pairs and clear the board.

Rules:

Player choose a pair of identical tiles to connect, and the connection path must not exceed two turns. If the requirements are met, two tiles will be eliminated. The player's task is to eliminate all the tiles on the board. The game is over if future moves are not possible and tiles are left on the board, or the player successfully removes all the tiles.

Some legal patterns of connection path are shown as follow:



To simplify the game, we start the game with a 6*6 board and only have 9 different symbols (represented by characters : '1', '2', '3', ..., '9').

Your quest is to determine whether the given pair is legal according to the rules shown above.

If it is legal, eliminate the pair and print out "pair matched" .

If it is not legal, do nothing and print out "bad pair" .

The status of the board (the remaining tiles) should always be updated when you receive a legal pair,

which means the same input might have different results over time.

Input :

[Beginning]

Given 6*6 characters to set all the initial symbol on the board.

[Loop]

Given a pair of coordinates on the board, represented by 4 integers (pos1.x pos1.y pos2.x pos2.y).

Output :

The result message of each given pair. (“bad pair” or “pair matched”)

Sample Input / Output :

Sample Input	Sample Output
9 2 4 8 9 7	bad pair
6 9 1 7 1 5	pair matched
5 3 3 7 2 2	pair matched
3 4 7 1 9 5	pair matched
4 6 8 8 2 6	pair matched
6 5 1 4 3 8	bad pair
2 1 3 3	pair matched
3 1 3 2	
2 1 3 3	
5 1 5 3	
1 1 4 3	
0 1 5 4	
3 0 3 4	

- ☐ Easy, only basic programming syntax and structure are required.
- ☒ Medium, multiple programming grammars and structures are required.
- ☐ Hard, need to use multiple program structures or complex data types.

Expected solving time:

35 minutes

Other notes:

The connection path is not restricted by the size of the board.