Day 22   new

September LeetCoding Challenge

Premium

Description | Solution | Discuss (999+) | Submissions

C++   Autocomplete

4. Any dead cell with exactly three live neighbors becomes a live cell, as if by reproduction.

Write a function to compute the next state (after one update) of the board given its current state. The next state is created by applying the above rules simultaneously to every cell in the current state, where births and deaths occur simultaneously.

Example:

```
Input:
[
  [0,1,0],
  [0,0,1],
  [1,1,1],
  [0,0,0]
]
Output:
[
  [0,0,0],
  [1,0,1],
  [0,1,1],
  [0,1,0]
]
```

Follow up:

1. Could you solve it in-place? Remember that the board needs to be updated at the same time: You cannot update some cells first and then use their updated values to update other cells.

2. In this question, we represent the board using a 2D array. In principle, the board is infinite, which would cause problems when the active area encroaches the border of the array. How would you address these problems?

Accepted 192,446   Submissions 348,604

Seen this question in a real interview before?   Yes   No

Companies 🔒   ⌄

Related Topics   ⌄

Similar Questions   ⌄

```cpp
class Solution {
public:
    int numNeighbours(vector<vector<int>>& v, int i, int j, int m, int n){
        int x=i,y=j;
        int ans=0;
        for(int ii=-1;ii<=1;ii++){
            for(int jj=-1;jj<=1;jj++){
                if(ii || jj){
                    x=i+ii;
                    y=j+jj;
                    if(x>=0 && x<m && y>=0 && y<n){
                        ans+=v[x][y]&1;
                    }
                }
            }
        }
        return ans;
    }
    void gameOfLife(vector<vector<int>>& v) {
        if(v.empty())
            return;
        int m=v.size();
        int n=v[0].size();
        for(int i=0;i<m;i++){
            for(int j=0;j<n;j++){
                int a=numNeighbours(v,i,j,m,n);
                cout<<a<<" ";
                if(a==3 && v[i][j]==0){
                    v[i][j]=1<<1;
                }
                else if(a==3 || a==2){
                    int g=v[i][j];
                    v[i][j]=v[i][j]<<1;
                    v[i][j]=v[i][j]|g;
                }
            }
            cout<<"\n";
        }
        for(int i=0;i<m;i++){
            for(int j=0;j<n;j++){
                v[i][j]=v[i][j]>>1;
            }
        }
    }
};
```

Your previous code was restored from your local storage. Reset to default

Problems   Pick One   Prev   🐞/59   Next   Console ⌄   Contribute *i*   ▶ Run Code   Submit