package com.my.algocode;

import java.util.Arrays;

import java.util.HashMap;

import java.util.Map;

import java.util.Scanner;

public class Test03 {

    private static int[][] arr;

    private static Map<Integer,Character> map1 = new HashMap<>();

    private static Map<Integer,Map<Integer,Integer>> map2 = new HashMap<>();

    private static int[][] result;

    private static int flag = 0;

    public static void main(String[] args) {

        Scanner input = new Scanner(System.in);

        int n = input.nextInt();

        char[] color = new char[n];

        for(int i=0;i<n;i++){

            color[i] = input.next().charAt(0);

        }

        for(int i=0;i<n;i++){

            map1.put(i,color[i]);

        }

        arr=new int[n][6];

        result=new int[n][6];

        for(int i=0;i< result.length;i++){

            for(int j=0;j<result[i].length;j++){

                result[i][j]=-1;

            }

        }

        for(int i=0;i<n;i++){

            Map<Integer,Integer> tempMap = new HashMap<>();

            for(int j=0;j<6;j++){

                arr[i][j]=input.nextInt();

                tempMap.put(j,arr[i][j]);

            }

            map2.put(i,tempMap);

        }

        dfs(0,n);

    }

    private static void dfs(int t,int n){

        if(flag==1){

            return;

        }

        if(t==n){

            for(int i=0;i<result.length;i++){

                for(int j=0;j<result[i].length;j++){

                    System.out.print(map1.get(result[i][j]));

                }

                System.out.println();

            }

            flag=1;

            return;

        }

        else {

            for(int j=0;j<4;j++){

                for(int k=0;k<t-1;k++) {

                    if (result[k][j]==result[k+1][j]) {

                        return;

                    }

                }

            }

            for(int i=0;i<6;i++){

                result[t]=findColor(i,t);

                dfs(t+1,n);

                result[t]=new int[]{-1,-1,-1,-1,-1,-1};

            }

        }

    }

    private static int[] findColor(int x,int t){

        int[] colorNew = new int[6];

        if(x==0){

            int[] location = {0,1,2,3,4,5};

            for(int i=0;i<6;i++){

                colorNew[i]=map2.get(t).get(location[i]);

            }

        }

        else if(x==1){

            int[] location = {1,0,3,2,4,5};

            for(int i=0;i<6;i++){

                colorNew[i]=map2.get(t).get(location[i]);

            }

        }

        else if(x==2){

            int[] location = {2,3,1,0,4,5};

            for(int i=0;i<6;i++){

                colorNew[i]=map2.get(t).get(location[i]);

            }

        }

        else if(x==3){

            int[] location = {3,2,0,1,4,5};

            for(int i=0;i<6;i++){

                colorNew[i]=map2.get(t).get(location[i]);

            }

        }

        else if(x==4){

            int[] location = {4,5,2,3,1,0};

            for(int i=0;i<6;i++){

                colorNew[i]=map2.get(t).get(location[i]);

            }

        }

        else {

            int[] location = {5,4,2,3,0,1};

            for(int i=0;i<6;i++){

                colorNew[i]=map2.get(t).get(location[i]);

            }

        }

        return colorNew;

    }

}