Laboratory work # 7

Student: HU Riqian
Student ID: 20321114
Timus Name: hduads2022_20321114
Mail: jhlxhrq@163.com

Problem # 1080. Map Coloring

Screenshot from Timus:

9874528	14:16:11 15 May 2022	hduads2022_20321114	1080. Map Coloring	Java 1.8	Accepted		0.125	784 KB

Explanation of algorithm:

1. It is the famous Map Coloring problem related to the famous four-color theorem, which was the first theorem to be proved primarily by a computer.

2. We use the DFS algorithms can easily work out the problem.

Computational complexity of algorithm:

$$O(N^2)$$

Source code:

```
import java.io.*;

public class MapColoring {
    public static void main(String[] args) throws IOException {
        new MapColoring().run();
    }

    StreamTokenizer in;
    PrintWriter out;

    int nextInt() throws IOException {
        in.nextToken();
        return (int) in.nval;
    }

    void run() throws IOException {
        in = new StreamTokenizer(new BufferedReader(new
InputStreamReader(System.in)));
        out = new PrintWriter(System.out);
        solve();
```

```
        out.flush();
    }

    int[][] MAP = new int[100][100];
    int[] COLOR = new int[100];
    static boolean isInvalid = false;
    static int NUMBER;

    void print(int x, int c) {
        if (!isInvalid) {
            if (COLOR[x] != 2) {
                if (COLOR[x] != c) {
                    isInvalid = true;
                }
                return;
            }
            COLOR[x] = c;
            for (int i = 0; i < NUMBER; i++) {
                if (MAP[i][x] != 0) {
                    MAP[i][x] = MAP[x][i] = 0;
                    print(i, 1 - c);
                }
            }
        }
    }

    void solve() throws IOException {
        NUMBER = nextInt();
        int i = 0;
        int t;
        while (i < NUMBER) {
            COLOR[i] = 2;
            t = nextInt();
            if (t != 0) {
                MAP[t - 1][i] = MAP[i][t - 1] = 1;
            } else {
                i++;
            }
        }
        for (int j = 0; j < NUMBER; j++) {
            if (COLOR[j] == 2) {
                print(j, 0);
            }
        }
        outputResult();
    }

    void outputResult() {
        String result = "";
        if (isInvalid) {
            result = "-1";
        } else {
            for (int j = 0; j < NUMBER; j++) {
                result += COLOR[j];
            }
        }
        out.println(result);
    }
}
```