

# 状压dp练习

---

## 人员

---

左子毅、朱奕鸣、杨洋、刘子淇、赵清航、周子航、于迦浩、穆鹏宇、程晟泰、梁钰涵、胡赫轩、崔嘉睿 到课

## 作业

---

<https://www.luogu.com.cn/contest/168604>

A、B题必做，C选做

<https://www.luogu.com.cn/contest/162467>

刷题题单要求同学们课下进行练习

## 课堂表现

---

课堂整体课堂纪律不错，课上题目有些难度，同学们课下要好好总结复习

## 课堂内容

---

### Bicolorings

$f[i][j][k]$ : 第 $i$ 列，状态为 $j$ ，有 $k$ 个联通块的情况数

$f[i][1][k] += f[i-1][0][k-1] \dots$

共有16种转移情况，可以预先写一个  $4*4$  的数组代表 16 种转移情况

```

#include <bits/stdc++.h>

using namespace std;

const int maxn = 1000 + 5;
const int mod = 998244353;
int f[maxn][4][maxn*2]; // f[i][j][k]: i行,状态j,有k个联通块的方案数

int value(int l, int j) {
    if (l == j) return 0;
    if (l==0 || l==3) return 1;
    if (j==0 || j==3) return 0;
    return 2;
}

int main()
{
    int n, m; cin >> n >> m;
    f[1][0][1] = f[1][3][1] = f[1][1][2] = f[1][2][2] = 1;
    for (int i = 2; i <= n; ++i) {
        for (int j = 0; j < 4; ++j) {
            for (int k = 0; k <= m; ++k) {
                for (int l = 0; l < 4; ++l) {
                    if (k-value(l,j) >= 0) f[i][j][k] = (f[i][j][k] + f[i-1][l][k-value(l,j)]) %
mod;
                }
            }
        }
    }

    int res = 0;
    for (int i = 0; i < 4; ++i) res = (res + f[n][i][m]) % mod;
    cout << res << endl;
    return 0;
}

```

## Kim's Quest

$f[1][j]$ : 长度为1, 状态为j的子序列的总数  
 $f[2][j]$ : 长度为2, 状态为j的子序列的总数  
 $f[3][j]$ : 长度 $\geq 3$ , 状态为j的子序列的总数

```

#include <bits/stdc++.h>

using namespace std;

typedef long long LL;
const int maxn = 2e5 + 5;
const int mod = 998244353;
int w[maxn], f[4][8];

int iValue(int x, int y) { return x*2 + y; }
int iValue(int x, int y, int z) { return x*4 + y*2 + z; }

int main()
{
    int n; cin >> n;
    for (int i = 1; i <= n; ++i) cin >> w[i], w[i] &= 1;

    for (int i = 1; i <= n; ++i) {
        int a = f[2][iValue(0,0)] + f[3][iValue(0,0,0)];
        int b = f[2][iValue(0,1)] + f[3][iValue(1,0,1)];
        int c = f[2][iValue(1,0)] + f[3][iValue(1,1,0)];
        int d = f[2][iValue(1,1)] + f[3][iValue(0,1,1)];
        if (!w[i]) {
            f[3][iValue(0,0,0)] = ((LL)f[3][iValue(0,0,0)] + a) % mod;
            f[3][iValue(1,1,0)] = ((LL)f[3][iValue(1,1,0)] + d) % mod;
        } else {
            f[3][iValue(0,1,1)] = ((LL)f[3][iValue(0,1,1)] + b) % mod;
            f[3][iValue(1,0,1)] = ((LL)f[3][iValue(1,0,1)] + c) % mod;
        }

        f[2][iValue(0, w[i])] = (f[2][iValue(0, w[i])] + f[1][0]) % mod;
        f[2][iValue(1, w[i])] = (f[2][iValue(1, w[i])] + f[1][1]) % mod;

        f[1][w[i]]++;
    }

    int res = 0;
    res = (res + f[3][iValue(0,0,0)]) % mod;
    res = (res + f[3][iValue(1,1,0)]) % mod;
    res = (res + f[3][iValue(0,1,1)]) % mod;
    res = (res + f[3][iValue(1,0,1)]) % mod;
    cout << res << endl;
    return 0;
}

```