

## 12.一些真题 || 模拟题

### 1、最少次数

有一个整数  $A=2021$ ，每一次，可以将这个数加 1、减 1 或除以 2，其中除以 2 必须在数是偶数的时候才允许。

例如，2021 经过一次操作可以变成 2020、2022。

再如，2022 经过一次操作可以变成 2021、2023 或 1011。

请问，2021 最少经过多少次操作可以变成 1。

```
1 //暴力
2 #include<iostream>
3 using namespace std;
4
5 int main() {
6     int s = 0;
7     int n = 2021;
8     while (n != 1) {
9         if (n % 10 == 1 || n % 10 == 5 || n % 10 == 9) {
10             n--;
11             s++;
12         }
13         if (n % 10 == 3 || n % 10 == 7) {
14             n++;
15             s++;
16         }
17         if (n % 2 == 0) {
18             n /= 2;
19             s++;
20         }
21     }
22     cout << s;
23     return 0;
24 }
```

### 2、螺旋矩阵

```
1 class Solution {
2 public:
3     vector<vector<int>> generateMatrix(int n) {
4         vector<vector<int>> mat(n,vector<int> (n));
5         int l,r,b,t,num;
6         l = t = 0;
7         r = b = n-1;
```

```

8         num = 1;
9         while(num <= n*n)
10        {
11            for (int i = l; i<= r;i++)    mat[t][i]=num++;
12                t++;
13            for (int i = t; i <= b; i++)    mat[i][r]=num++;
14                r--;
15            for (int i = r; i >= l;i--)    mat[b][i]=num++;
16                b--;
17            for (int i = b; i >= t;i--)    mat[i][l]=num++;
18                l++;
19        }
20        return mat;
21    }
22 };

```

### 3、动态规划——爬楼梯

```

1  #include <iostream>
2  #include <algorithm>
3  using namespace std;
4  int f[100005]; //f[i] : 走到第i层的最小消耗
5  int cost[100005];
6  int n;
7  int main()
8  {
9      f[0] = f[1] = 0;
10     cin >> n;
11     for (int i = 0; i < n; i++)
12     {
13         cin >> cost[i];
14     }
15     for (int i = 2; i <= n; i++)    //i=2到n
16     {
17         f[i] = min(f[i-1]+cost[i-1],f[i-2]+cost[i-2]);
18     }
19     cout << f[n];
20     return 0;
21 }

```

### 4、

```

1  #include <iostream>
2  #include <algorithm>
3  using namespace std;
4
5  int ans[20200005];
6
7
8  int main()
9  {

```

```
10     ans[1] = ans[2] = ans[3] = 1;
11     for (int i = 4; i<=20190324;i++)
12     {
13         ans[i]=ans[i-1]%10000+ans[i-2]%10000+ans[i-3]%10000;
14         ans[i]%=10000;
15         //cout << ans[i]<<endl;
16     }
17
18     cout << ans[20190324];
19     return 0;
20 }
```