

## 2/26 题解

### A 单词游戏

较为复杂的 dfs，要维护目标串的下标和是否能拐弯，写两个方向数组保证转弯一定是直角而不是  $45^\circ$

同时需要写 8 个方向的坐标，因为可以走斜线

### B 三角形

由题可知，一共有两行三角形。

首先遍历所有三角形判断周围三个面有多少面不与其他三角形重合  
(计算偶数列的三角形时只需判断左右两个三角形是否与它相邻)

```
11 ans=0;
for(int i=1;i<=2;i++){
    for(int j=1;j<=n;j++){
        if(a[i][j]==1){
            if(j%2==0){
                11 k=3;
                if(a[i][j-1]==1)k--;
                if(a[i][j+1]==1)k--;
                ans+=k;
            }
            if(j%2==1){
                11 k=3;
                if(a[i-1][j]==1)k--;
                if(a[i][j-1]==1)k--;
                if(a[i+1][j]==1)k--;
                if(a[i][j+1]==1)k--;
                ans+=k;
            }
        }
    }
}
```

最后统计总个数,输出 **ans** 即可

## C 构成等差数列

设等差数列的首项为  $a_1$  末项为  $a_n$ , 则有

$n = \frac{(a_1 + a_n)(a_n - a_1 + 1)}{2}$  = 等差数列的和.

$2n = (a_1 + a_n)(a_n - a_1 + 1)$

(1)  $a_1$  为奇数,  $a_n$  为奇数

则  $a_1 + a_n$  为偶数,  $a_n - a_1 + 1$  为奇数

(2)  $a_1$  为偶数,  $a_n$  为奇数

则  $a_1 + a_n$  为奇数,  $a_n - a_1 + 1$  为偶数

(3)  $a_1$  为奇数,  $a_n$  为偶数

则  $a_1 + a_n$  为奇数,  $a_n - a_1 + 1$  为偶数

(4)  $a_1$  为偶数,  $a_n$  为偶数

则  $a_1 + a_n$  为偶数,  $a_n - a_1 + 1$  为奇数

综上,  $a_1 + a_n$  与  $a_n - a_1 + 1$  奇偶性不同

所以分解  $2n$  的因数, 设  $ab = n$

当  $a$  和  $b$  奇偶性不同时

1.  $a_1 + a_n = a$ ,  $a_n - a_1 - 1 = b$

2.  $a_n - a_1 - 1 = a$ ,  $a_1 + a_n - a_n - a_1 - 1 = b$

$ans += 2$

最后输出  $ans$ .

核心代码:

```
ll x;
cin >> x;
x *= 2;
ll ans = 0;
for (ll i = 1; i <= sqrt(x); i++) {
    if (x % i == 0 && (i + x / i) % 2 == 1) ans += 2;
}
cout << ans << endl;
```

## D. 搭配礼物

比较水, 用 `map` 储存在数组 `b` 中 `m - a[i]` 的下标 (如果有), 然后循环判断两个下标是否满足下标差, 满足就 `ans++`

注意 `long long`

还好不重复，不然这个方法不行

核心: (bl 是 map)

```
for (long long i=1;i<=n;i++)cin>>a[i];
for (long long i=1;i<=n;i++)cin>>b[i],bl[b[i]]=i;
for (long long i=1;i<=n;i++){
    long long left=m-a[i];
    if (bl[left]!=0&&abs(bl[left]-i)>=k)ans++;
}
cout<<ans;
```