Minimum Swaps for Bracket Balancing

pos记录所有的 [的位置

每当count < 0，说明当前的] 需要跟下一个 [进行交换（利用pos可以很方便的进行交换）

```cpp
long swapCount(string s)
{
    // Keep track of '['
    vector<int> pos;
    for (int i = 0; i < s.length(); ++i)
        if (s[i] == '[')
            pos.push_back(i);

    int count = 0; // To count number of encountered '['
    int p = 0;  // To track position of next '[' in pos
    long sum = 0; // To store result

    for (int i = 0; i < s.length(); ++i)
    {
        if (s[i] == '[')
        {
            ++count;
            ++p;
        }
        else if (s[i] == ']')
            --count;

        if (count < 0)
        {
            // printf("pos[p] = %d, i =%d\n", pos[p], i);
            sum += pos[p] - i;
            swap(s[i], s[pos[p]]);
            ++p;
            count = 1;
        }
    }
    return sum;
}
```

括号匹配类的题目，主要思路在于记录左括号的位置