

-基础知识积累

栈/队列

做题练习巩固,如: ([牛客网](#)) 吐泡泡 (栈, 队列)

```
1  #include<bits/stdc++.h>
2  using namespace std;
3
4  int main()
5  {
6      string s;
7      while(cin >> s)
8      {
9          vector<char> t;
10         for(int i = 0; i<s.size(); i++)
11         {
12             if(s[i] == 'o')
13             {
14                 if(!t.empty()&&t.back()=='o')
15                 {
16                     t.pop_back();
17                     if(!t.empty()&&t.back()=='O') t.pop_back();
18                     else {
19                         t.push_back('O');
20                     }
21                 }
22                 else {
23                     t.push_back('o');
24                 }
25             }
26             if(s[i] == 'O')
27             {
28                 if(!t.empty()&&t.back()=='O') t.pop_back();
29                 else {
30                     t.push_back('O');
31                 }
32             }
33         }
```

```
34         for(auto x : t) cout << x;
35         cout << endl;
36     }
37
38     return 0;
39 }
```

图论

作业:

```
1  #include<bits/stdc++.h>
2  using namespace std;
3
4  const int N = 100000;
5  int n, m;
6  bool a[N]; //判断是否阅读
7  vector<int> v[N]; //利用邻接表存储
8  queue<int> step;
9
10 void dfs(int x, int count){
11     cout << x << " ";
12     a[x]=true;
13     if(count==n) return;
14     for(int i = 0; i<v[x].size(); i++)
15     {
16         if(!a[v[x][i]]) dfs(v[x][i], count+1);
17     }
18 }
19
20 void bfs(){
21
22     step.push(1);
23
24     a[1] = true;
25     while(!step.empty())
26     {
```

```

25         int b = step.front();
26         step.pop();
27         cout << b << " ";
28         for(int l = 0; l<v[b].size(); l++) {
29             if(!a[v[b][l]]){
30                 a[v[b][l]]=true;
31                 step.push(v[b][l]);
32             }
33         }
34     }
35 }
36
37 int main()
38 {
39     cin >> n >> m;
40     for(int i = 0; i<m; i++){
41         int x, y;cin >> x >> y;
42         v[x].push_back(y);
43     } //将关系存入邻接表
44     for(int i = 1; i<=n; i++) sort(v[i].begin(), v[i].end());
45     memset(a, false, sizeof(a)); //初始化路径
46     dfs(1, 0);
47     cout << "\n";
48     memset(a, false, sizeof(a)); // 初始化路径
49     bfs();
50
51     return 0;
52 }

```

小程序

图像 (image,view)的使用

以及滑动组件 (scroll-view) 的使用

-常用方法积累

```
1 bool Isprime(long long int n) {           //快速判断素数
2     for (int i = 2; i <= (long long int)sqrt(n); i++) {
3         if (n % i == 0) {
4             return false;
5             break;
6         }
7     }
8     return true;
9 }
```

```
1 #include<bits/stdc++.h>
2 using namespace std;
3
4 int main()    //已知长方体一个顶点三个面积求周长
5 {
6     int a, b, c;cin >> a >> b >> c;
7     int sum;
8     double x1, x2, x3;
9     x1 = sqrt((a*c)/b);
10    x2 = sqrt((b*a)/c);
11    x3 = sqrt((c*b)/a);
12    sum = 4*(x1+x2+x3);
13
14    cout << sum;
15
16    return 0;
17 }
```

```
1 find(v.begin(), v.end(), key);//用于查询数组元素是否存在
2 next_permutation(v.begin(), v.end());//用于数组全排列
```

