欧拉回路/通路

Euler Circuit/Path

Concept：图 G 的欧拉回路是包含 G 中每一条边的简单回路；图 G 的欧拉通路是包含 G 中每一条边的简单通路。（简单：不重复走边，即经过所有边恰好一次）

Theorem：

- 无向图存在欧拉回路当且仅当该图没有奇度顶点；
- 无向图存在欧拉通路当且仅当该图有且仅有 2 个奇度顶点；
- 有向图存在欧拉回路当且仅当该图所有点出度等于入度；
- 有向图存在欧拉通路当且仅当该图仅有 2 个点出度不等于入度，且这两个点其一入度比出度大 1，另一出度比入度大 1.

Heirholzer Algorithm

Idea：dfs 一遍，每次回溯时把当前点加入答案；无需打 `vis` 标记，不断删边即可。

Complexity：$O(V+E)$

Code（`vector` 存边）：

```
vector<pii> edge[N];
// edge[i].first is the vertex;
// edge[i].second is the id of this edge

bool vis[M];
stack<int> sta;
void dfs(int x){
    while(!edge[x].empty()){
        pii to = edge[x].back(); edge[x].pop_back();
        if(vis[to.second])  continue;
        vis[to.second] = true; // mark the edge
        dfs(to.first);
    }
    sta.push(x);
}
int main(){
    read(n, m);
    for(int i = 1; i <= m; i++){
        int u, v; read(u, v);
        edge[u].pb(mp(v, i));
        edge[v].pb(mp(u, i));
    }
    int st = 1;
    for(int i = 1; i <= n; i++){
        if(edge[i].size() & 1){
            st = i;
            break;
        }
    }
    dfs(st);
    while(!sta.empty()){
        printf("%d\n", sta.top());
        sta.pop();
    }
    return 0;
}
```