

# Tarjan求割边代码实现



主讲人：邓哲也



# 代码实现

我们用邻接表来存边。

```
struct edge {  
    int v, next;  
} e[M];  
  
int h[N], ee, dfn[N], low[N], tot, vis[N], n, m;  
  
void addedge(int u, int v) {  
    e[ee] = (edge) {v, h[u]};  
    h[u] = ee ++;  
}
```

# 代码实现

Tarjan算法主体部分:

```
void tarjan(int u, int fa) {
    vis[u] = 1;
    dfn[u] = low[u] = ++ tot;
    for (int i = h[u]; i != -1; i = e[i].next) {
        int v = e[i].v;
        if (!vis[v]) {
            tarjan(v, u);
            low[u] = min(low[u], low[v]);
            if (low[v] > dfn[u]) {
                printf("%d %d\n", u, v);
            }
        } else if (v != fa) {
            low[u] = min(low[u], dfn[v]);
        }
    }
}
```

# 代码实现

调用Tarjan算法:

```
for(int i = 1;i <= n;i ++)  
    if(!vis[i])  
        tarjan(i, 0);
```

# 代码实现

在这张图上调用 Tarjan 算法：

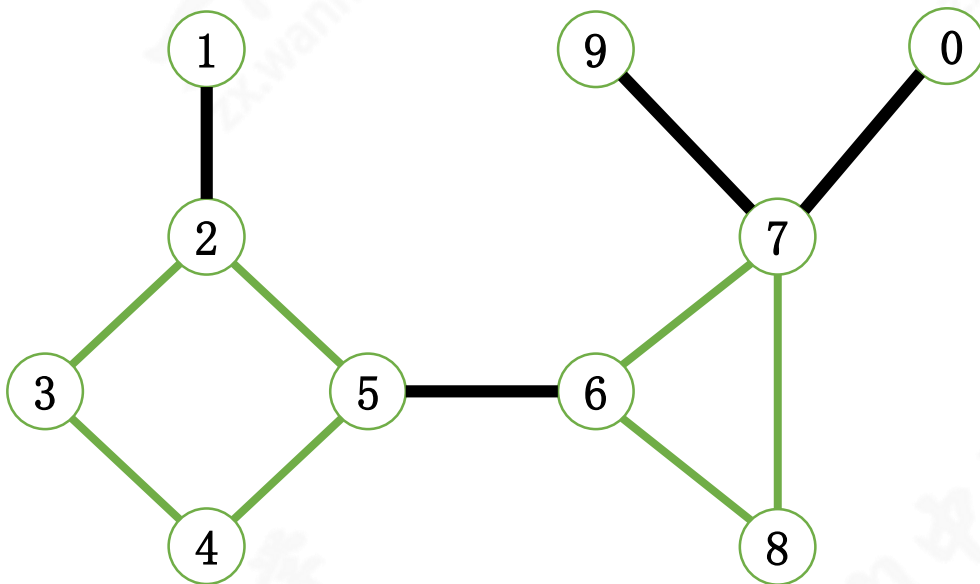
输出：

7 10

7 9

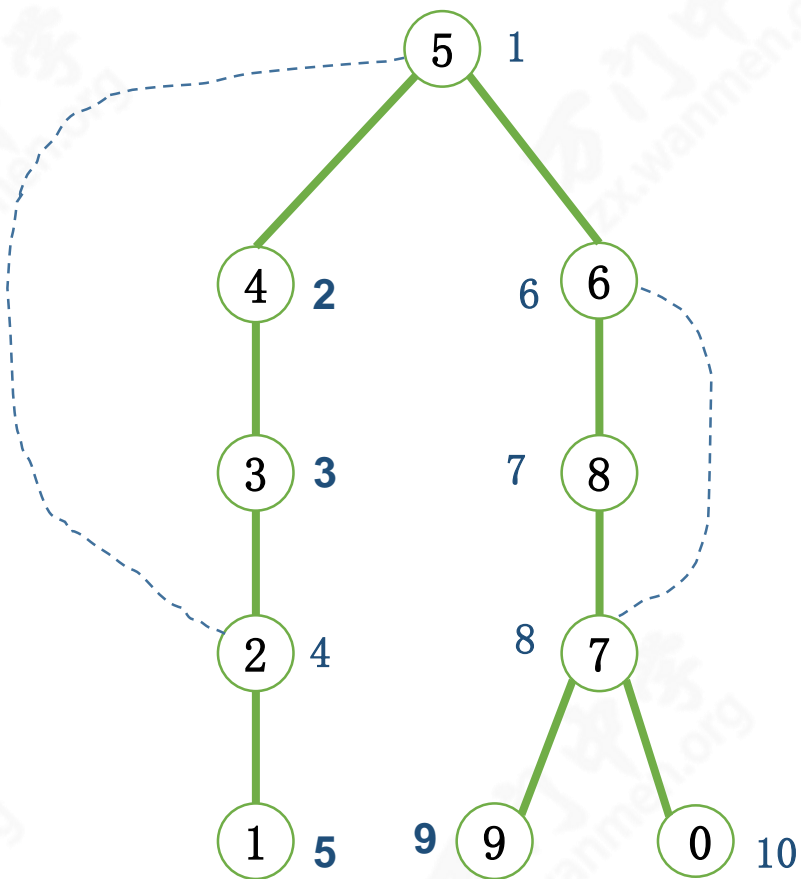
5 6

2 1



# 代码实现

调用 Tarjan(5)



	1	2	3	4	5	6	7	8	9	0
dfn	5	4	3	2	1	6	8	7	9	10
low	5	1	1	1	1	6	6	6	9	10

下节课再见