vector <**int**> adj[10010];  
**int** tme, cnt, d[10010], low[10010], par[10010];  
  
**void** init(){  
 tme=cnt=0;  
 memset(d, 0, **sizeof**(d));  
 memset(par, -1, **sizeof**(par));  
 **for**(**int** i=0; i<10010; i++) adj[i].clear();  
}  
  
**void** dfs(**int** nd, **bool** isRoot){  
 tme++;  
 d[nd]=tme;  
 low[nd]=d[nd];  
 **bool** fg=**false**;  
 **int** childcnt=0;  
 **for**(**int** i=0; i<adj[nd].size(); i++){  
 **int** v=adj[nd][i];  
 **if**(par[v]==-1 && v!=par[nd]){  
 par[v]=nd;  
 childcnt++;  
 dfs(v, **false**);  
 **if**(low[v]>=d[nd]){  
 fg=**true**;  
 }  
 low[nd]=min(low[nd], low[v]);  
 }  
 **if**(v!=par[nd]){  
 low[nd]=min(low[nd], d[v]);  
 }  
 }  
 **if**(fg && !isRoot){  
 cnt++;  
 printf("%d is an articulation point**\n**", nd);  
 }  
 **if**(isRoot){  
 **if**(childcnt>1) cnt++;  
 printf("%d is an articulation point**\n**", nd);  
 }  
}