

并查集



主讲人：邓哲也



大纲

并查集

并查集的操作

并查集的应用

并查集的实现

并查集

- 并查集保持一组不相交的动态集合

$S = \{S_1, S_2, \dots, S_k\}$ 。

- 每个集合通过一个代表来识别。
- 代表就是集合中的某个成员。
- 哪一个成员被选做代表是无所谓的。
- 我们关心的是如果寻找某一动态集合的代表两次，并且在两次寻找之间不修改集合，两次得到的答案应该是相同的。

并查集的操作

Make-Set (x)

建立一个新的集合，其唯一成员，也就是其代表，就是 x 。

因为各集合是不相交的，故要求 x 没有在其他集合中出现过。

Union(x, y)

将包含 x 和 y 的动态集合合并为一个新的集合（即这两个集合的并集）。

假定在这个操作之前两个集合是不相交的。然后选出一个代表。

Find-Set (x)

返回一个指针，指向包含 x 的（唯一）集合代表

并查集的应用

简单应用：统计一个无向图中连通子图的个数。

并查集的应用

简单应用：统计一个无向图中连通子图的个数。

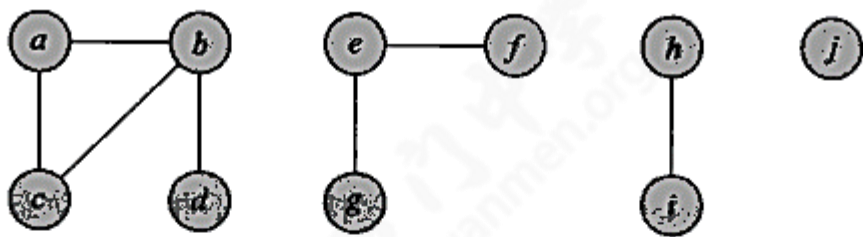
对于每个 V 中的顶点 v ， $\text{Make-Set}(v)$

对于每个 E 中的边 (u, v) ，如果 $\text{Find-Set}(u) \neq \text{Find-Set}(v)$ ，

则 $\text{Union}(u, v)$

最后统计有几个集合即可。（统计代表的个数）

并查集的应用



a)

处理的边	不相交集组									
初始集合	{a}	{b}	{c}	{d}	{e}	{f}	{g}	{h}	{i}	{j}
(b, d)	{a}	{b, d}	{c}		{e}	{f}	{g}	{h}	{i}	{j}
(e, g)	{a}	{b, d}	{c}		{e, g}	{f}		{h}	{i}	{j}
(a, c)	{a, c}	{b, d}			{e, g}	{f}		{h}	{i}	{j}
(h, i)	{a, c}	{b, d}			{e, g}	{f}		{h, i}		{j}
(a, b)	{a, b, c, d}				{e, g}	{f}		{h, i}		{j}
(e, f)	{a, b, c, d}				{e, f, g}			{h, i}		{j}
(b, c)	{a, b, c, d}				{e, f, g}			{h, i}		{j}

b)

并查集的实现

用有根树来表示集合

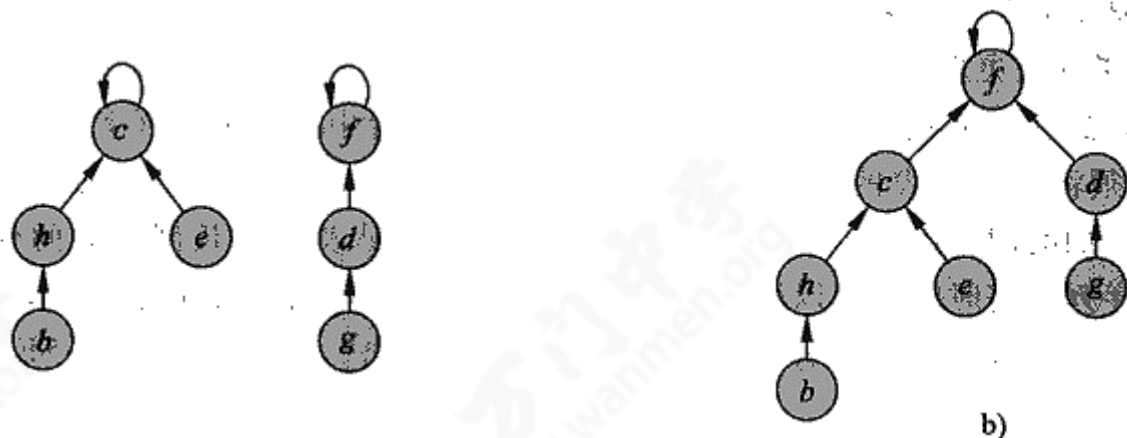
树中的每个节点都表示集合的一个成员

每棵树表示一个集合。

每个成员仅指向其父节点。

每棵树的根表示代表，且它是它自己的父节点。

思考之前的三个操作分别对应树上的什么操作？



并查集的实现

Make-Set

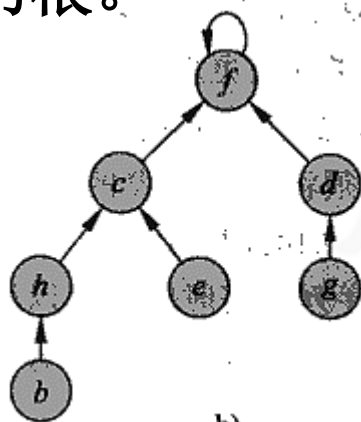
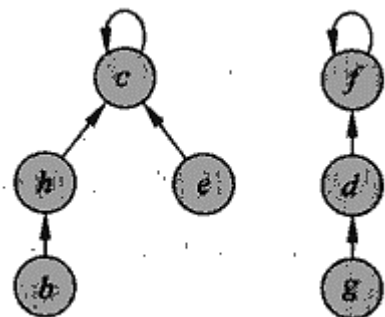
创建一颗仅包含一个节点的树

Find-Set

沿着父节点指针一直找下去，直到找到树根为止

Union

使得一颗树的根指向另一棵树的根。



b)

并查集的实现

Find-Set

沿着父节点指针一直找下去，直到找到树根为止

注意到一个包含 $n - 1$ 次 Union 操作的序列可能会构造出一颗为 n 个节点的线性链的树。

如何优化呢？

下节课再见