

本节内容

图的存储

十字链表、
邻接多重表

知识总览

图的存储

邻接矩阵

邻接表

十字链表

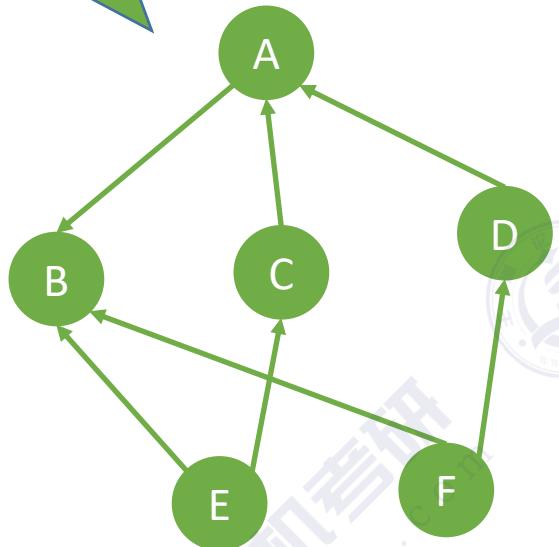
邻接多重表

存储有向图

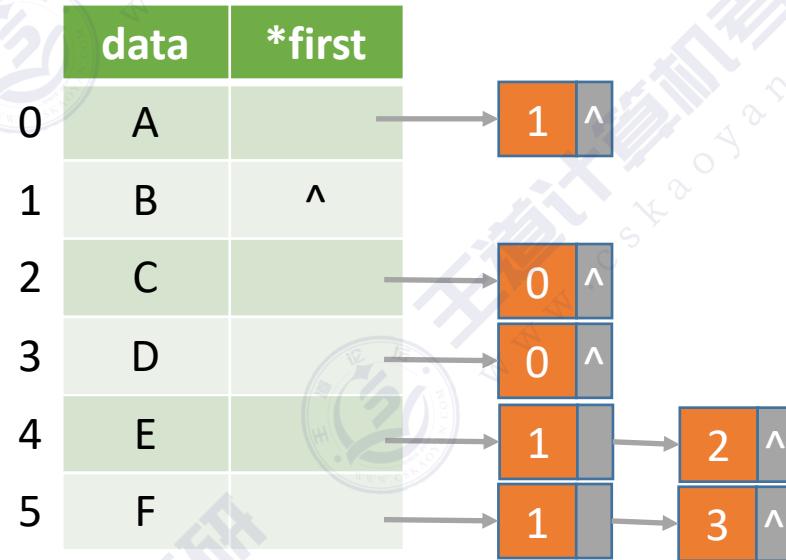
存储无向图

邻接矩阵、邻接表存储有向图

有向图



	A	B	C	D	E	F
A	0	1	1	1	0	0
B	1	0	0	0	1	1
C	1	0	0	0	1	0
D	1	0	0	0	0	1
E	0	1	1	0	0	0
F	0	1	0	1	0	0



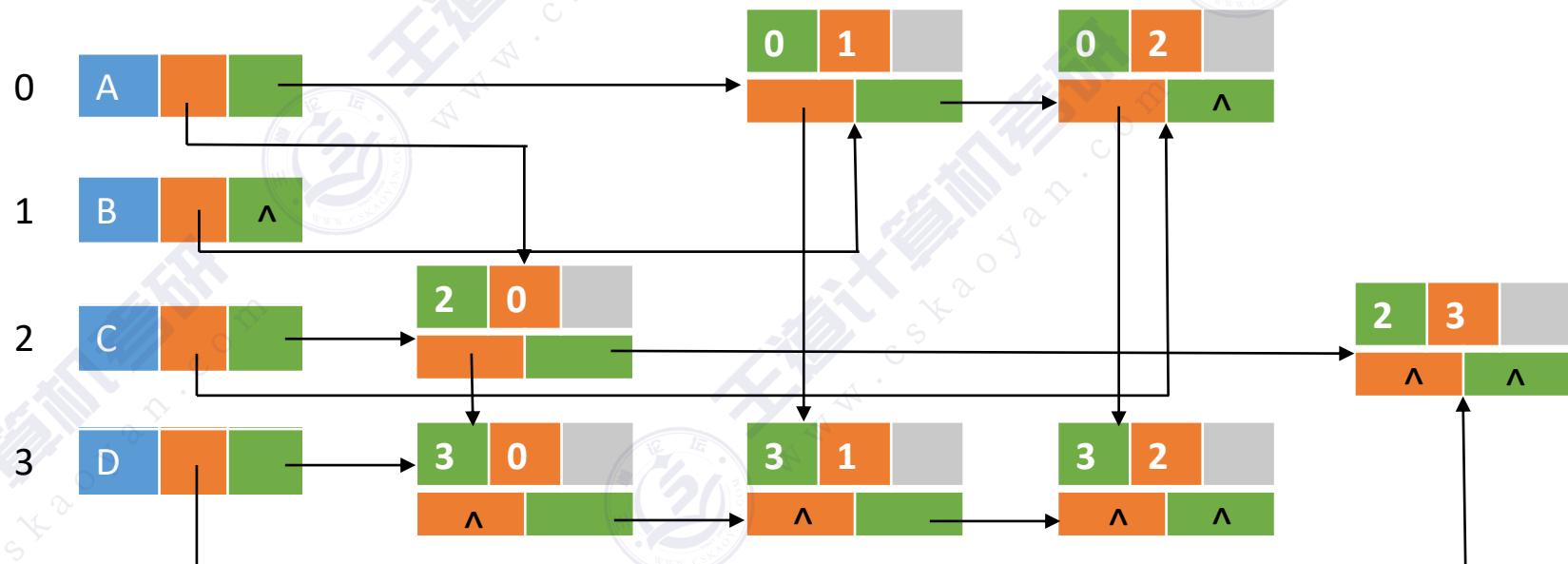
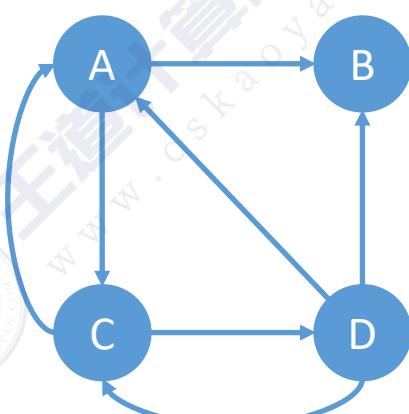
找顶点的入边不方便

邻接表

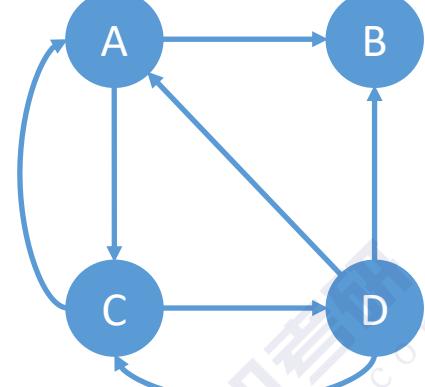
空间复杂度
高 $O(|V|^2)$

	邻接表	邻接矩阵
空间复杂度	无向图 $O(V + 2 E)$ ； 有向图 $O(V + E)$	$O(V ^2)$
计算度/出度/入度	计算有向图的度、入度不方便，其余很方便	必须遍历对应行或列
找相邻的边	找有向图的入边不方便，其余很方便	必须遍历对应行或列

十字链表存储有向图



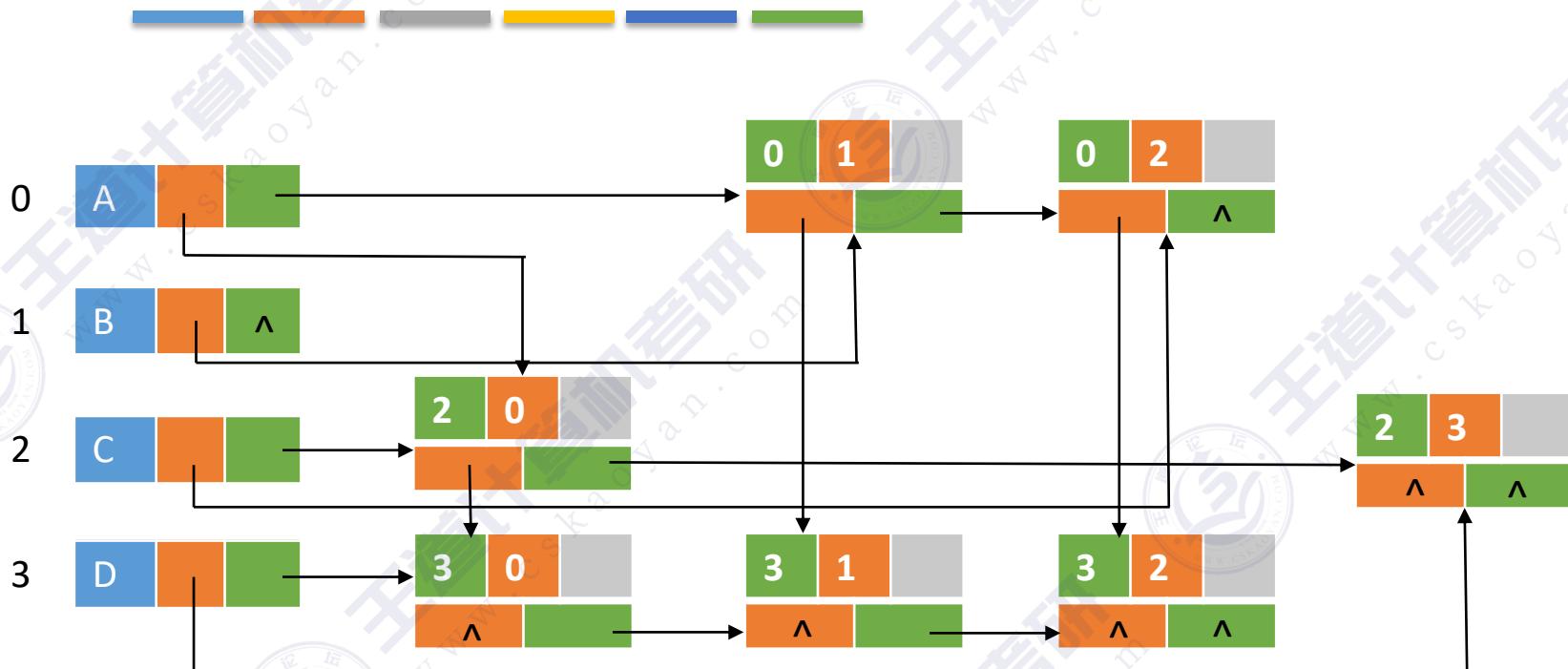
十字链表法性能分析



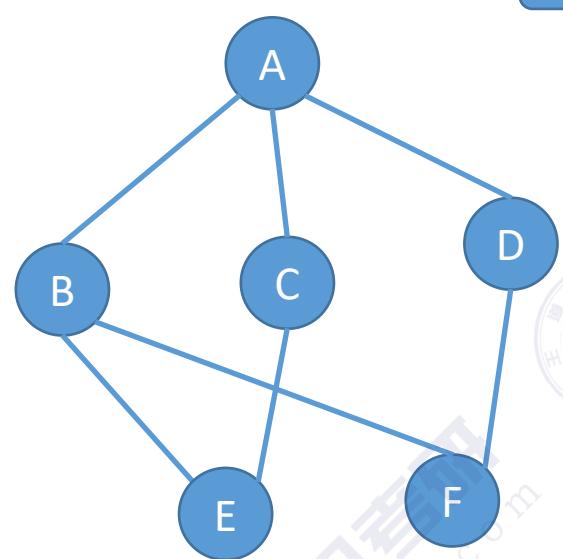
空间复杂度: $O(|V|+|E|)$

如何找到指定顶点的所有出边? ——顺着绿色线路找
如何找到指定顶点的所有入边? ——顺着橙色线路找

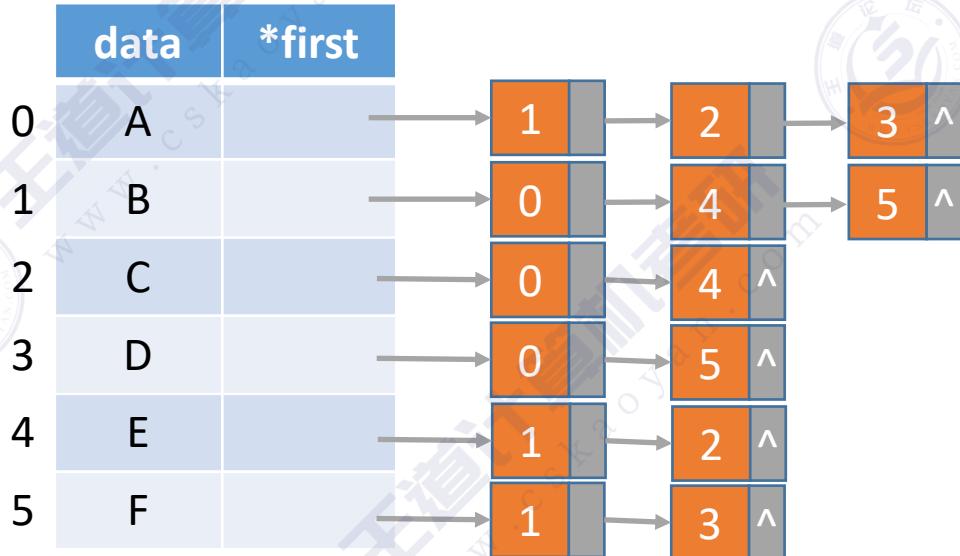
注意: 十字链表只用于存储有向图



邻接矩阵、邻接表存储无向图



邻接表



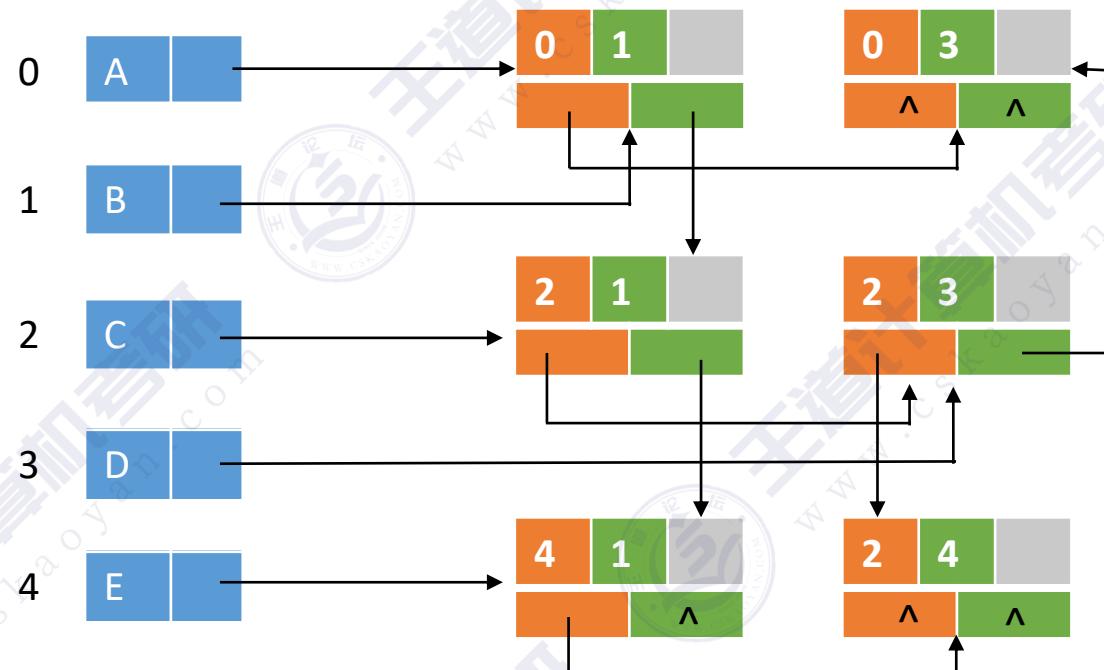
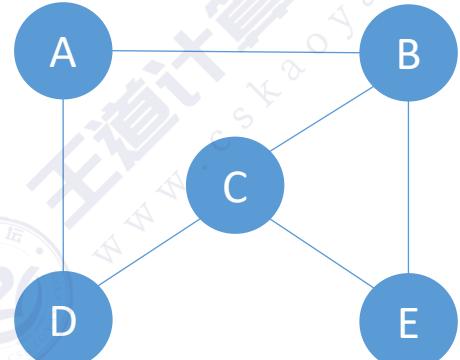
邻接矩阵

	A	B	C	D	E	F
A	0	1	1	1	0	0
B	1	0	0	0	1	1
C	1	0	0	0	1	0
D	1	0	0	0	0	1
E	0	1	1	0	0	0
F	0	1	0	1	0	0

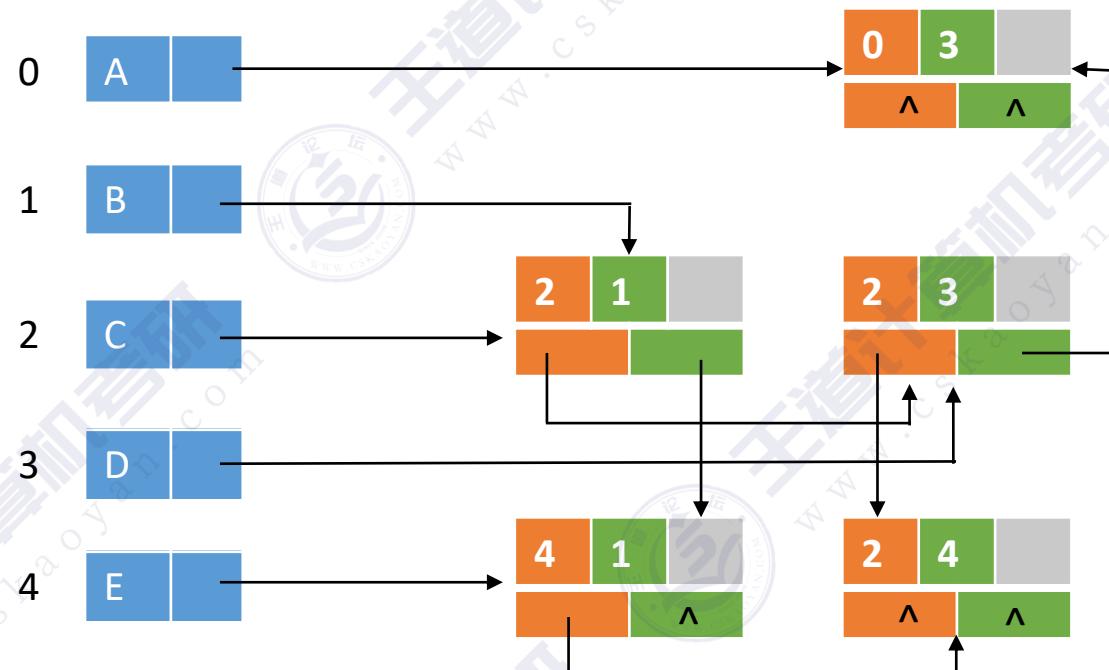
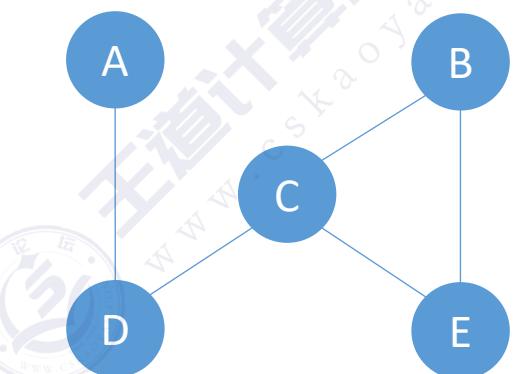
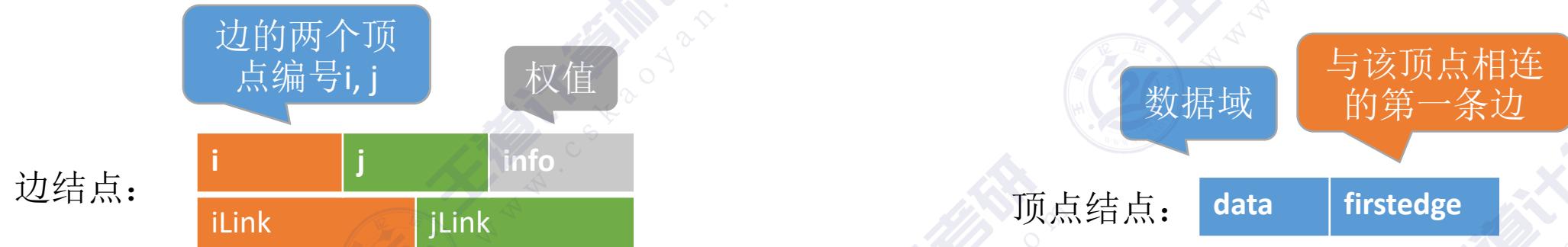
每条边对应两份冗余信息，
删除顶点、删除边等操作
时间复杂度高

空间复杂度高
 $O(|V|^2)$

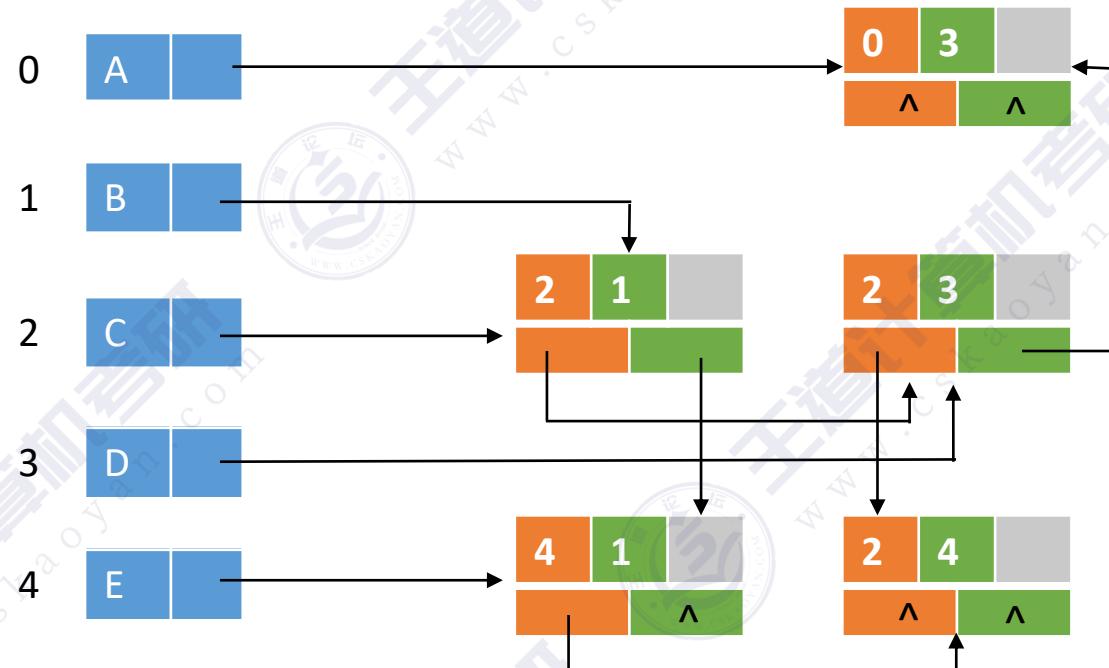
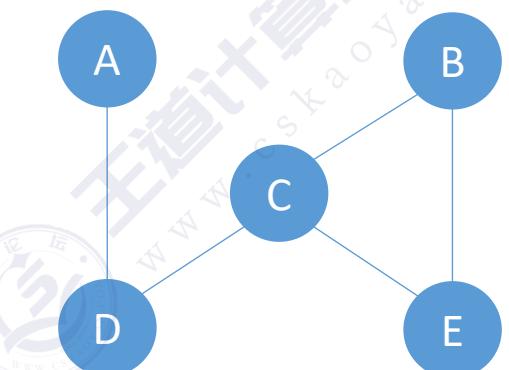
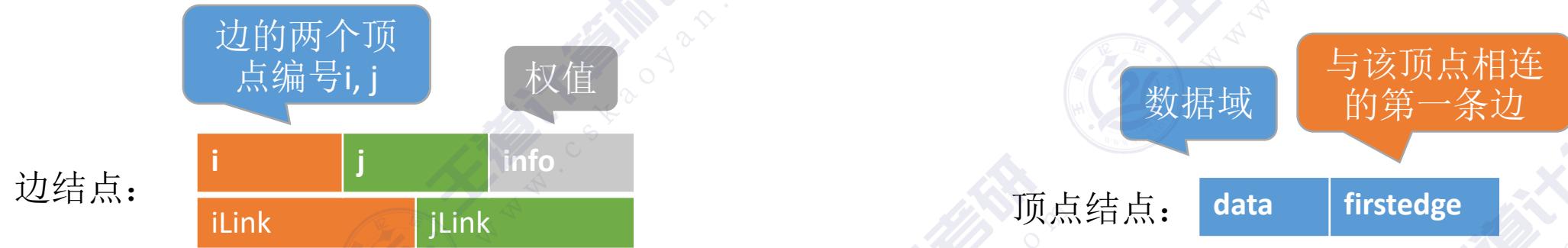
邻接多重表存储无向图



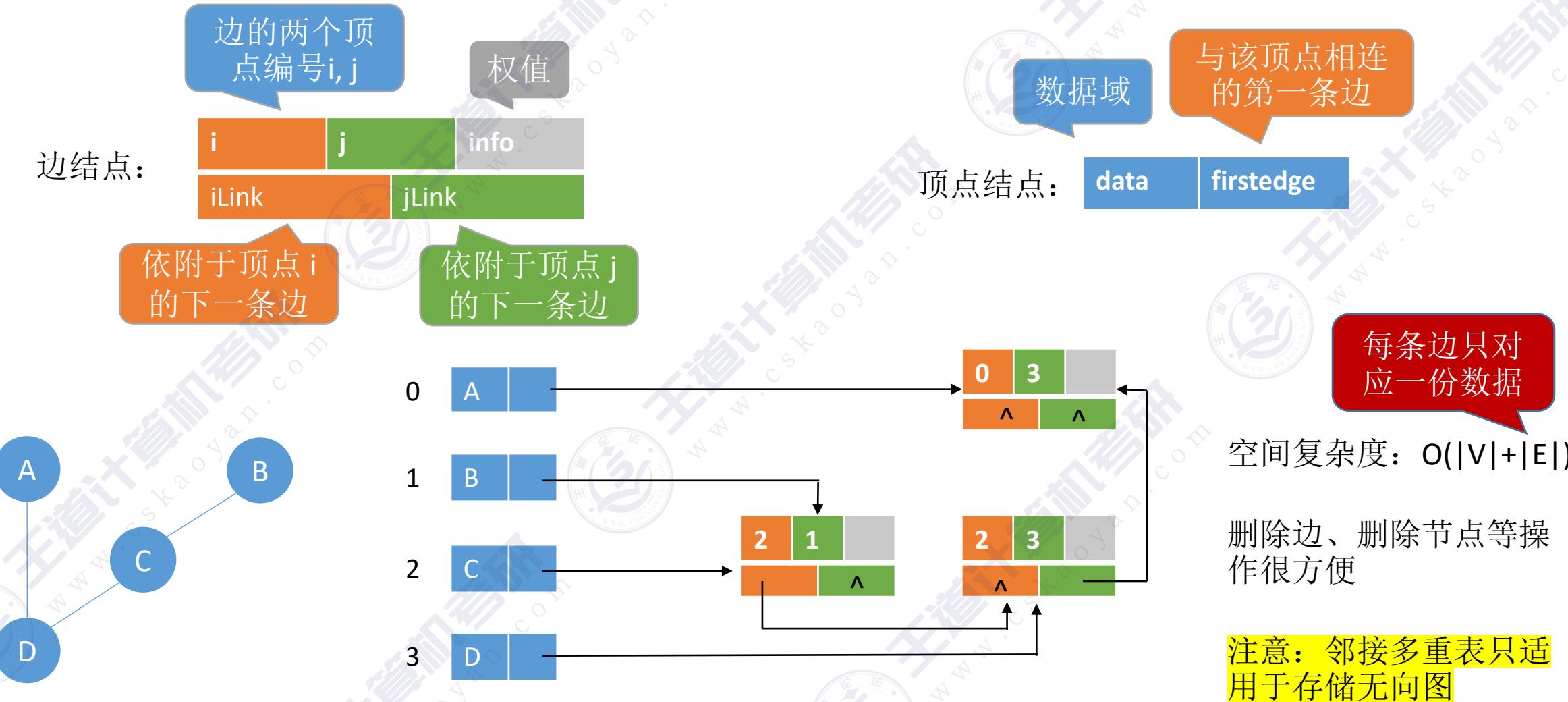
邻接多重表存储无向图



邻接多重表存储无向图



邻接多重表存储无向图



知识回顾与重要考点

	邻接矩阵	邻接表	十字链表	邻接多重表
空间复杂度	$O(V ^2)$	无向图 $O(V + 2 E)$ 有向图 $O(V + E)$	$O(V + E)$	$O(V + E)$
找相邻边	遍历对应行或列 时间复杂度为 $O(V)$	找有向图的入边必须遍历整个邻接表	很方便	很方便
删除边或顶点	删除边很方便，删除顶点需要大量移动数据	无向图中删除边或顶点都不方便	很方便	很方便
适用于	稠密图	稀疏图和其他	只能存有向图	只能存无向图
表示方式	唯一	不唯一	不唯一	不唯一

弧结点:



十字
链表

顶点结点:



边结点:



邻接多
重表

顶点结点:

