



在线性表中，数据元素之间仅有线性关系，每个数据元素仅有一个直接前驱和一个直接后继；

在树型结构中，数据元素之间有着明显的层次关系，并且每一层上的数据元素可能和下一层多个元素（即其孩子结点）相关，但只能和上一层中的一个元素（即其双亲结点）相关；

而在图形结构中，结点之间的关系可以是任意的，图中任意两个数据元素之间都可能相关；

```
ADT Graph{
    数据对象 V: V是具有相同特性的数据元素的集合，称为顶点集

    数据关系 R:
    R = {VR}
    VR = {<v,w>|v,w∈V 且 P(v,w), <v,w>表示从v到w的弧，谓词P(v,w)定义了弧<v,w>的意义或信息}

    基本操作 P:
    //创建图
    Status CreateGraph(Graph);

    //销毁图
    Status DestroyGraph(Graph);

    //定位顶点的位置
    int LocateVertex(Graph, Vertex);

    //根据值获取顶点
    Vertex GetVertex(Graph, ElementType);

    //修改顶点的值
    Status PutVertex(Graph, Vertex, ElementType);

    //获取顶点的第一个邻接顶点
    Vertex FirstAdjacencyVertex(Graph, Vertex);

    //获取顶点的某个邻接顶点的下一个邻接顶点
    Vertex NextAdjacencyVertex(Graph, Vertex, Vertex);

    //插入顶点
    Status InsertVertex(Graph, Vertex);

    //删除顶点
    Status DeleteVertex(Graph, Vertex);

    //插入边
    Status InsertArc(Graph, Vertex, Vertex);

    //删除边
    Status DeleteArc(Graph, Vertex, Vertex);
}
```

```
//深度优先遍历
void DFSTraverse(Graph);

//广度优先遍历
void BFSTraverse(Graph);
}
```

中文名称	英文名称	作用
顶点	Vertex	在图中的数据元素通常称作顶点
V	英文名称	V是顶点的有穷非空集合
VR	英文名称	VR是两个顶点之间的关系的集合
-	-	-
弧	Arc	若 $\langle v,w \rangle \in VR$ ，则 $\langle v,w \rangle$ 表示从v到w的一条弧
弧尾	Tail	接上，且称v是弧尾
弧头	Head	接上，且称w是弧头
有向图	Digraph	可以理解为弧是具有方向的边，此时的图称为有向图
-	-	-
边	Edge	若 $\langle v,w \rangle \in VR$ ，则必有 $\langle w,v \rangle \in VR$ ，即VR是对称的；则以无序对 (v,w) 表示这两个有序对，表示v和w之间的一条边
无向图	Undigraph	图的VR是对称的，边是没有方向的；此时的图称为无向图
-	-	-
完全图	Completed graph	有 $n(n-1)/2$ 条边的无向图称为完全图
有向完全图	-	有 $n(n-1)$ 条边的有向图称为有向完全图
-	-	-
稀疏图	Sparse graph	有很少条边或者弧的图称为稀疏图
稠密图	Dense graph	接上，反之称为稠密图
-	-	-
网	Network	这种带权的图称为网

中文名称	英文名称	作用
子图	Subgraph	$G = (V, \{E\})$ $G' = (V', \{E'\})$ 如果 $V' \subseteq V$ 且 $E' \subseteq E$, 则称 G' 为 G 的子图
-	-	-
邻接点	Adjacent	对于无向图 $G=(V, \{E\})$, 如果边 $(v, v') \in E$, 则称顶点 v 和 v' 互为邻接点, 即 v 和 v' 邻接
依附	Incident	边 (v, v') 依附于顶点 v 和 v' , 或者说 (v, v') 和顶点 v 与 v' 相关联
-	-	-
度	Degree	顶点 v 的度是和 v 相关联的边的数目, 记为 $TD(v)$
入度	In Degree	顶点 v 的度是以 v 为弧头的边的数目, 记为 $ID(v)$
出度	Out Degree	顶点 v 的度是以 v 为弧尾的边的数目, 记为 $OD(v)$
-	-	-
路径	Path	从顶点 v 到顶点 w 的路径是一个顶点序列 $(v=v_0, v_1, \dots, v_m, \dots, w)$ 如果 G 是有向图, 则路径也是有方向的
环、回路	Cycle	第一个顶点和最后一个顶点相同的路径称为回路或环
简单路径	-	路径序列中顶点不重复出现的路径称为简单路径
-	-	-
连通	Connected	如果从顶点 v 到顶点 v' 有路径, 则称 v 和 v' 是连通的
连通图	Connected Graph	在无向图 G 中, 如果对于图中任意两个顶点 $v_i, v_j \in V$, v_i 和 v_j 都是连通的, 则称 G 是连通图
连通分量	Connected Component	连通分量是指无向图中的极大连通子图 (加上任何额外的结点都会导致不连通)
-	-	-
强连通图	Strongly Connected Graph	在有向图 G 中, 如果对于图中任意两个顶点 $v_i, v_j \in V$, v_i 和 v_j 都是连通的, 则称 G 是强连通图
强连通分量	Strongly Connected Component	强连通分量是指有向图中的极大连通子图 (加上任何额外的结点都会导致不连通)
-	-	-
生成树	Spanning Tree	一个连通图的生成树是一个极小连通子图, 它含有图中全部顶点, 但只有足以构成一颗树的 $n-1$ 条边 (删除任何一条边都会导致不再连通)

中文名称	英文名称	作用
生成森林	Spanning Forest	含有图中全部顶点，但只有足以构成若干颗不相交的树的边（弧线）