图--21计科李明

- **定义/逻辑结构**
  - 定义
    - 图(Graph)是由顶点的有穷非空集合V (G) V(G)V(G)和顶点之间边的集合E (G) E(G)E(G)组成，通常表示为: G = (V , E)，其中，G表示个图，V是图G 中顶点的集合，E是图G中边的集合
  - 分类
    - 有向图
      - 若E是有向边(也称弧)的有限集合时，则图G为有向图
    - 无向图
      - 若E是无向边(简称边)的有限集合时，则图G为无向图
  - 相关术语与概念
    - 简单图
      - 不存在重复边;
      - 不存在顶点到自身的边
    - 多重图
      - 与简单图相对的
    - 完全图
      - 对于无向图，任意两个顶点之间存在边
      - 对于有向图，任意两个顶点之间都存在方向相反的两条弧
    - 子图
      - G'是G的子图：V'是V的自己，E'是E的子图
    - 连通
      - 连通的概念
        - 在无向图中，若从顶点v 到顶点w 有路径存在，则称v和w是连通的
      - 连通图
        - 图G 中任意两个顶点都是连通的
      - 强连通图
        - 在有向图中，若从顶点v 到顶点w和从顶点w到项点v之间都有路径,则称这两个顶点是强连通的
      - 非连通图
        - 图G 中存在两个顶点不是连通的
      - 连通分量
        - 无向图中的极大连通子图称为连通分量
      - 强连通分量
        - 有向图中的极大强连通子图称为有向图的强连通分量
    - 生成树/森林
      - 连通图的生成树是包含图中全部顶点的一个极小连通子图
      - 在非连通图中，连通分量的生成树构成了非连通图的生成森林
    - 顶点的度、入度和出度
      - 每个顶点的度定义为以该项点为一个端点的边的数目
      - 对于有向图,顶点v的度分为入度和出度,入度是以顶点v为终点的有向边的数目
    - 边的权和网
      - 在一个图中，每条边都可以标上具有某种含义的数值，该数值称为该边的权值。这种边上带有权值的图称为带权图，也称网。
    - 稠密图/稀疏图
      - 边数很少的图称为稀疏图，反之称为稠密图。稀疏和稠密本身是模糊的概念，稀疏图和稠密图常常是相对而言的。
    - 路径，路径长度和回路
      - 路径上边的数目称为路径长度。第一个顶点和最后一个顶点相同的路径称为回路或环
      - 路径：顶点v到p之间的一条路径是指顶点序列
      - 在路径序列中，顶点不重复出现的路径称为简单路径。除第一个顶点和最后一个顶点外，其余顶点不重复出现的回路称为简单回路。
- **存储结构**
  - 邻接矩阵
    - 定义/组成
      - 图的邻接矩阵 存储方式是用两个数组来表示图。一个一维数组存储图中顶点信息，一个二维数组(称为邻接矩阵)存储图中的边或弧的信息
    - 特点
      - 无向图的邻接矩阵一定是一个对称矩阵
  - 邻接表
    - 定义
      - 对图G中的每个顶点v 建立一个单链表，第i个单链表中的结点表示依附于顶点v的边这个单链表就称为顶点v的边表，边表组成一个线性表，称其为邻接表存图法
    - 组成
      - 顶点表结点由顶点域(data)和指向第一条邻接边的指针(firstarc) 构成，边表(邻接表)结点由邻接点域(adjvex)和指向下一条邻接边的指针域(nextarc) 构成。
  - 比较
    - 对于邻接表
      - 对于稀疏图，邻接表法节省存储空间
      - 便于增加和删除顶点。
      - 便于统计边的数目，时间复杂度是O(n+e)
    - 对于邻接矩阵
      - 方便查找某两边之间是否存在边
      - 方便寻找某一顶点直接相邻的点
      - 方便计算各个顶点的入度和出度
- **图的遍历**
  - 深度优先遍历
    - 概念/过程
      - 深度优先搜索类似于树的先序遍历：首先访问图中某一起始顶点v，然后由v出发，访问与v邻接且未被访问的任一顶点w再访问与w邻接且未被访问的任一顶点...重复上述过程。当不能再继续向下访问时，依次退回到最近被访问的顶点，若它还有邻接顶点未被访问过，则从该点开始继续上述搜索过程，直至图中所有顶点均被访问过为止。
    - 性能
      - DFS算法是一个递归算法，需要借助一个递归工作栈，故其空间复杂度为O (V)
      - 邻接矩阵时间复杂度:O(V^2)
      - 邻接表时间复杂度:O(V+E)
  - 广度优先遍历
    - 概念/过程
      - 类似于树的层序遍历：广度优先搜索是一种分层的查找过程，每向前走一步可能访问一批顶点，不像深度优先搜索那样有往回退的情况，因此它不是一个递归的算法。为了实现逐层的访问，算法必须借助一个辅助队列，以记忆正在访问的顶点的下一层顶点。
    - 性能
      - 无论是邻接表还是邻接矩阵的存储方式，BFS 算法都需要借助一个辅助队列Q, n个顶点均需入队一次，在最坏的情况下，空间复杂度为O (V)
      - 采用邻接表存储方式时，每个顶点均需搜索一次(或入队一次)， 在搜索任一顶点的邻接点时，每条边至少访问一次，算法总的时间复杂度为O(V + E)O(V+E)。采用邻接矩阵存储方式时，查找每个顶点的邻接点所需的时间为O (V，故算法总的时间复杂度为 O(V^2)
- **相关算法/应用**
  - 最下生成树
    - 算法
      - prime算法
        - 简单概括
          - 从一个顶点出发，在保证不形成回路的前提下，每找到并添加一条最短的边，就把当前形成的连通分量当做一个整体或者一个点看待，然后重复“找最短的边并添加”的操作
        - 复杂度
          - $O(n^2)$
      - kruskal算法
        - 简单概括
          - 与Prim算法从顶点开始扩展最小生成树不同，Kruskal 算法是一种按权值的递增次序选择合适的边来构造最小生成树的方法。
        - 复杂度
          - $O(n\log n)$
      - 克鲁斯卡尔算法主要是针对边来展开，边数少时效率会非常高，所以对于稀疏图有很大的优势；而普里姆算法对于稠密图，即边数非常多的情况会更好一些。
    - 概念
      - 一个连通图的生成树是一个极小的连通子图，它含有图中全部的顶点，但只有足以构成一棵树的n − 1条边，若砍去它的一条边，则会使生成树变成非连通图；若给它增加一条边，则会形成图中的一条回路。对于一个带权连通无向图G = (V , E)，生成树不同，其中边的权值之和最小的那棵生成树（构造连通网的最小代价生成树），称为G的最小生成树(Minimum-Spanning-Tree, MST)。
  - 最短路
    - 算法
      - dijkstra算法
        - 简单概括
          - 迪杰斯特拉(Dijkstra) 算法，它并不是一下子求出了v0到vi的最短路径，而是一步步求出它们之间顶点的最短路径，过程中都是基于已经求出的最短路径的基础上，求得更远顶点的最短路径，最终得到你要的结果。
        - 复杂度
          - $O(V^2)$
      - floyd算法
        - 简单概括
          - Floyd算法是一个迭代的过程，每迭代一次，在从vi到vj的最短路径上就多考虑了一个顶点；经n次迭代后所得到的就是vi到vj的最短路
        - 复杂度
          - $O(V^3)$
    - 概念
      - 无权图最短路径，其实就是指两顶点之间经过的边数最少的路径
      - 对于网图来说，最短路径，是指两顶点之间经过的边上权值之和最少的路径，并且我们称路径上的第一个顶点是源点，最后一个顶点是终点。
  - 拓扑排序
    - 相关概念
      - AOV网
        - 在一个表示工程的有向图中，用顶点表示活动，用弧表示活动之间的优先关系，这样的有向图为顶点表示活动的网，我们称为AOV网ActivityOnVertexNetwork)
      - 拓扑序列
        - 设G = (V , E)是一个具有n个顶点的有向图，V中的顶点序列V1 , V2 , . . . Vn，满足若从顶点Vi到Vj有一条路径，则在顶点序列中顶点Vi必在顶点Vj之前。则我们称这样的顶点序列为一个拓扑序列。
    - 算法实现
      - 每一轮选择一个入度为0的顶点并输出，然后删除该顶点和所有以它为起点的有向边，最后得到拓扑排序的结果
  - 关键路径
    - 相关概念
      - AOE网
        - 在带权有向图中，以顶点表示事件，以有向边表示活动，以边上的权值表示完成该活动的开销(如完成活动所需的时间)，称之为用边表示活动的网络，简称AOE网
      - 关键路径/关键活动
        - 在AOE网中仅有一个入度为0的顶点，称为开始顶点(源点)，它表示整个工程的开始；网中也仅存在一个出度为0的顶点，称为结束顶点(汇点)，它表示整个工程的结束。我们把路径上各个活动所持续的时间之和称为路径长度，从源点到汇点具有最大长度的路径叫关键路径，在关键路径上的活动叫关键活动。
    - 算法实现/过程
      - 从源点出发，零Ve(源点)=0，按拓扑排序求其余顶点的最早发生时间Ve()
      - 从汇点出发，令Vl(汇点)=Ve(汇点),按逆拓扑排序求其余顶点的最迟发生时间Vl()
      - 根据各顶点的Ve()值求所有弧的最早开始时间ee
      - 根据各顶点的Vl()值求所有弧的最迟开始时间el
      - 求AOE网中所有活动的差额(ee-el)，差额为0的构成关键路径