1. 课程简介
2. 图的基本概念
   1. 图的定义
   2. 顶点和边的定义
   3. 判断一个边是不是图
   4. 判断一个树是不是图
   5. 判断一个图是不是图
3. 图的分类
   1. 无向图 所有边没有方向
   2. 有向图 所有边都有方向
   3. 无向图中任意两个顶点存在边---无向完全图
      1. 性质，n个顶点，边为 n(n-1)/2
      2. 可以通过公式反推是不是无向完全图
   4. 有向图中的边，称为弧，弧头 弧尾的介绍
      1. 写公式时候 <v1,v2> v1是尾，v2是头
   5. 有向完全图，任意两个顶点存在方向互为相反的两条弧
      1. 性质 n个顶点， n(n-1)条边
4. 图的概念
   1. 权
   2. 网
   3. 稀疏图、稠密图
   4. 度
      1. 无向图度
      2. 有向图 入度 出度
   5. 邻接点
   6. 子图
   7. 路径的长度
5. 连通图生成树
6. 邻接矩阵概念
   1. 先铺垫树、栈、队列存储，再说邻接矩阵
7. 代码分析
8. 有向图分析和代码
9. 邻接表的存储结构分析
   1. 顶点用数组存
   2. 边用链表存
   3. 结构体要有三个
      1. 头节点结构体
      2. 邻接点结构体
      3. 图的结构体
   4. 邻接点的结构体
      1. int pos 顶点在顶点数组中的位置
      2. 指针域 指向下一个邻接点
   5. 头节点结构体
      1. char name[64] 节点的名字
      2. 指针域 指向第一个邻接点的地址
   6. 图的结构体
      1. 顶点个数
      2. 边的条数
      3. 顶点的结构体数组
10. 邻接表代码实现
11. 图的深度优先搜索
    1. DFS depth first search
    2. 重建一个数组和vertex顶点数组形成对应关系，标记该顶点是否被访问过
    3. 取一个没访问的顶点入栈，并且标记为访问，开始遍历
    4. 取栈顶元素邻接点，入栈，标记访问
    5. 没有邻接点了栈顶出栈，找新的栈顶元素邻接点入栈，标记访问