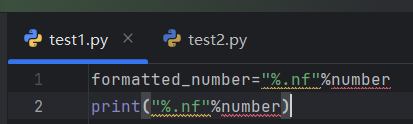
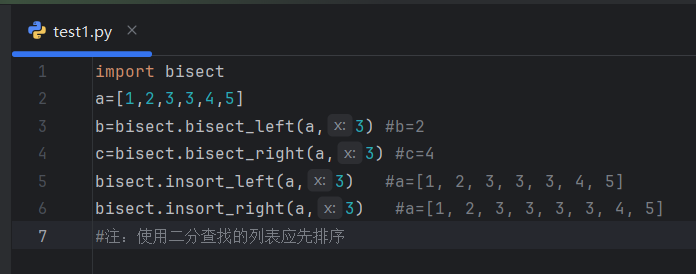
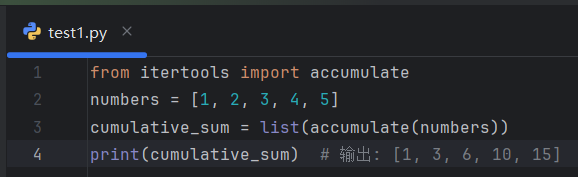
1.浮点数格式化：



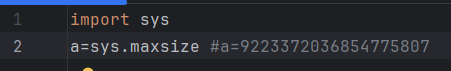
2.二分查找：



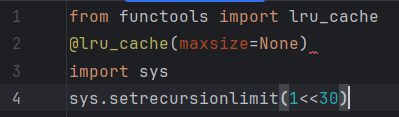
3.计算前缀和：



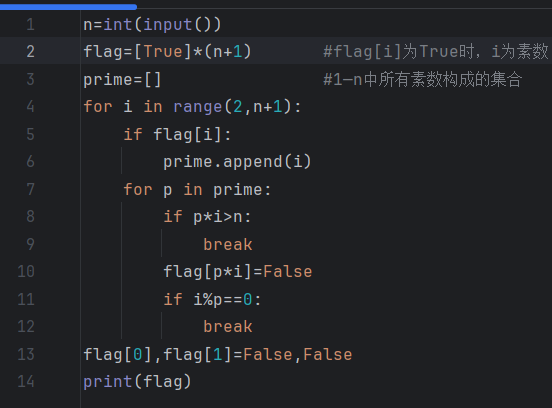
4.引入无穷大



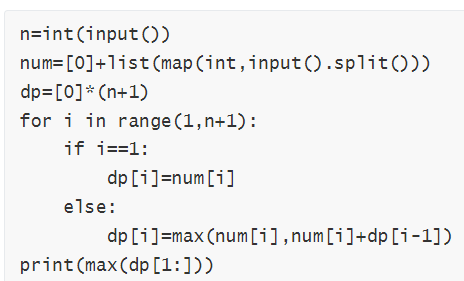
5.递归trick



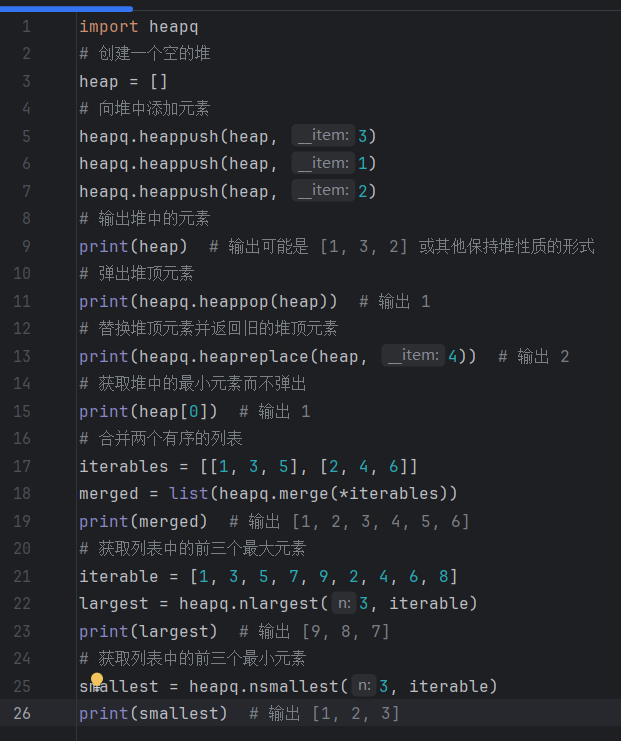
6，欧拉筛法



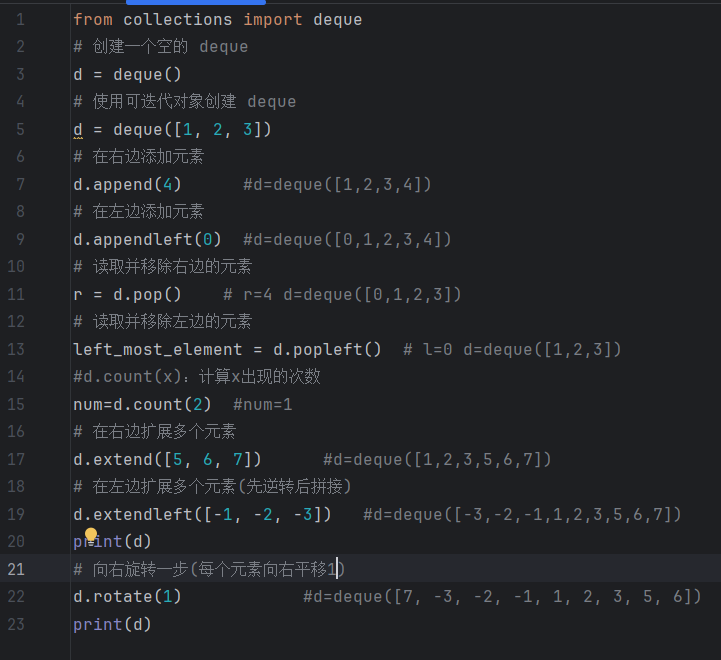
7. Kadane算法：在一个整数数组中找到一个具有最大和的连续子数组（至少包含一个数字）



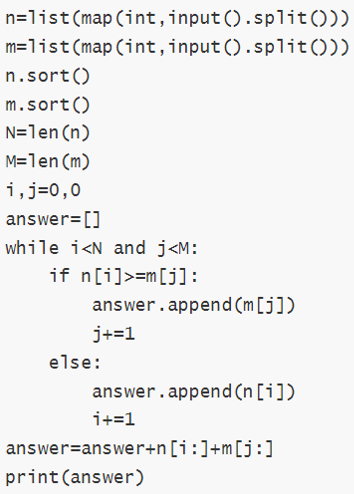
8．堆



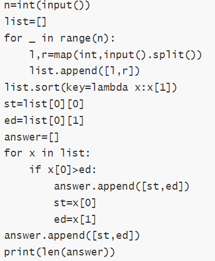
9.双端队列（deque）



10.递增数组合并：m,n为两个无序数组，合并为一个递增数组



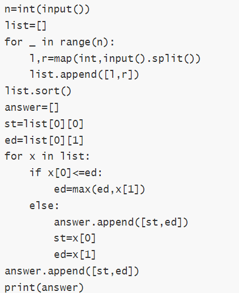
11．多个区间求最多的不相交区间



12.多个区间求最少分组使不交



13.多个区间取并集



1. 字符串：a.title():首字母大写 a.upper():全大写 a.lower():全小写

字符串掺入变量：f”…{a}…”

a.rstrip():删除右侧空白 a.lstrip():删除左侧空白 a.strip():删除两侧空白

2）列表：a.insert(索引，元素)：在索引处插入元素 del a[索引]：删除索引出元素

a.remove(元素)：删除a中第一个该元素(无则报错) a.reverse():翻转列表

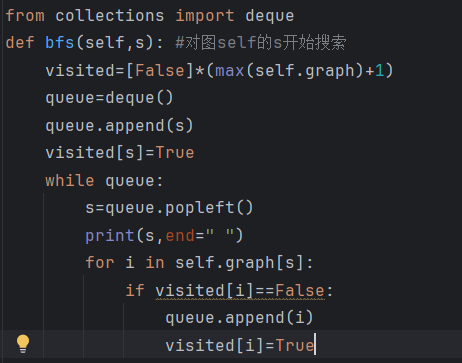
a.sort(reverse=True):永久性排序 b=sorted(a，reverse=True):临时排序

min(a):a的最小值 max(a):a的最大值 sum(a):a的总和

14.多个区间最少覆盖



15.bfs模板



3）字典：a={键：值,键：值} b=a.get(键，默认值)有该键则赋对应值，无则赋默认值

a[键]：访问对应值 a[键]=值：添加或修改键值对 del a[键]：删除键值对（无则报错）

for x,y in a.items():遍历键值对 for x in a.keys/values():遍历键/值 b=list(a.keys/values):导出键/值为列表

4）集合：a=set(列表)：将列表转化为集合 a={}:构建新集合 a.add():添加新元素

a.update(列表)：添加列表中多个元素 a.remove(元素)：删除元素（无则报错）

a.discard(元素):删除元素（无则不报错）b=a.pop():随机删除一个元素并赋于b

b.clear():清空b c=a.union(b):并集 c=a&b:交集 c=a-b:差集（a去掉b中有的元素）

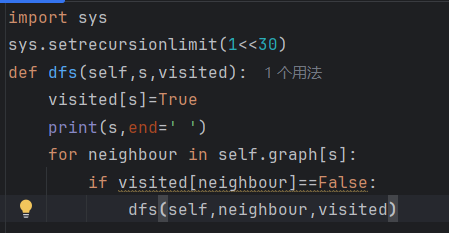
c=a^b:对称差集（并集减交集）a<=b:a是b的子集 a<b:a是b的真子集

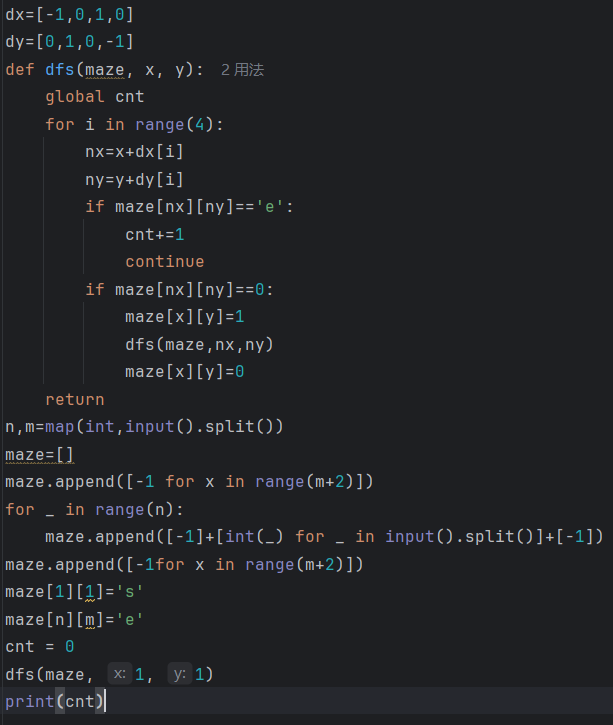
a.isdisjoint(b):a和b交集为空

5)while/for循环：break：退出循环 continue:结束本次循环进入下一次循环

6)函数：return:退出函数

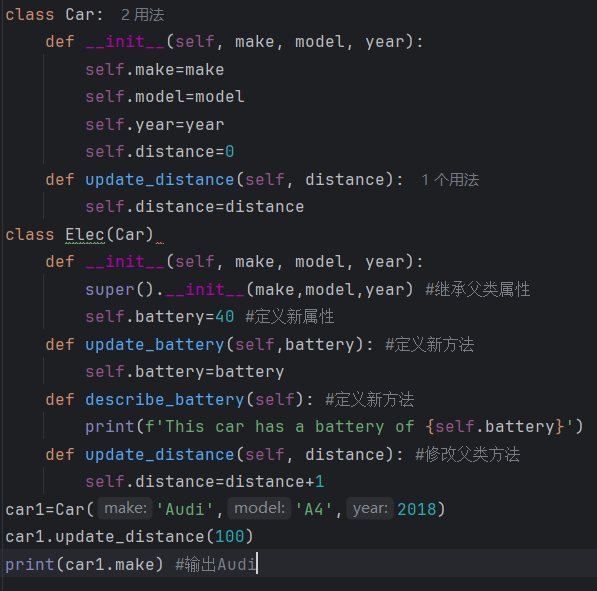
16.dfs模板





17.全局变量（见16图二的global）

18.类



19.树的遍历：

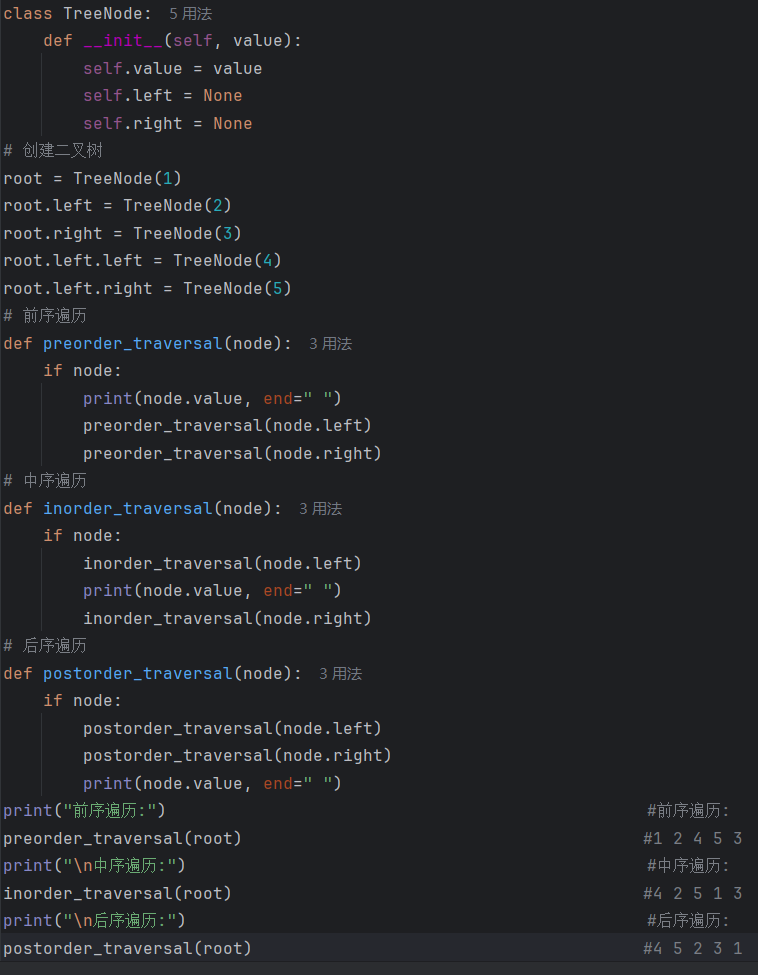
**深度优先遍历（DFS）**：

**前序遍历（Preorder）**：根 → 左 → 右

**中序遍历（Inorder）**：左 → 根 → 右（BST 中序遍历结果为有序数组）

**后序遍历（Postorder）**：左 → 右 → 根

**广度优先遍历（BFS）**（层序遍历）：按层从左到右依次遍历。



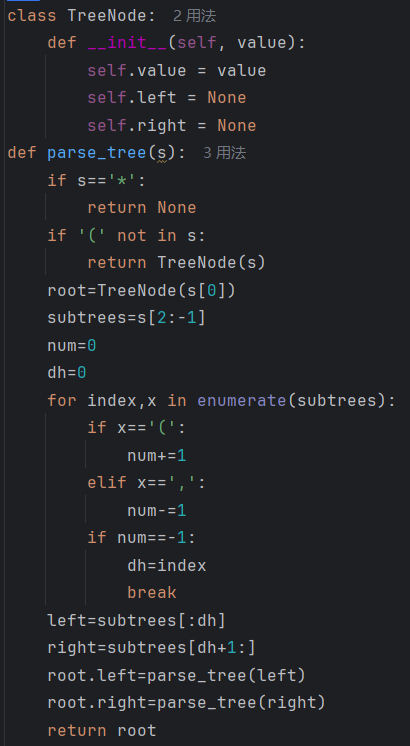
20. enumerate函数：

for index,x in enumerate(列表，数字)：

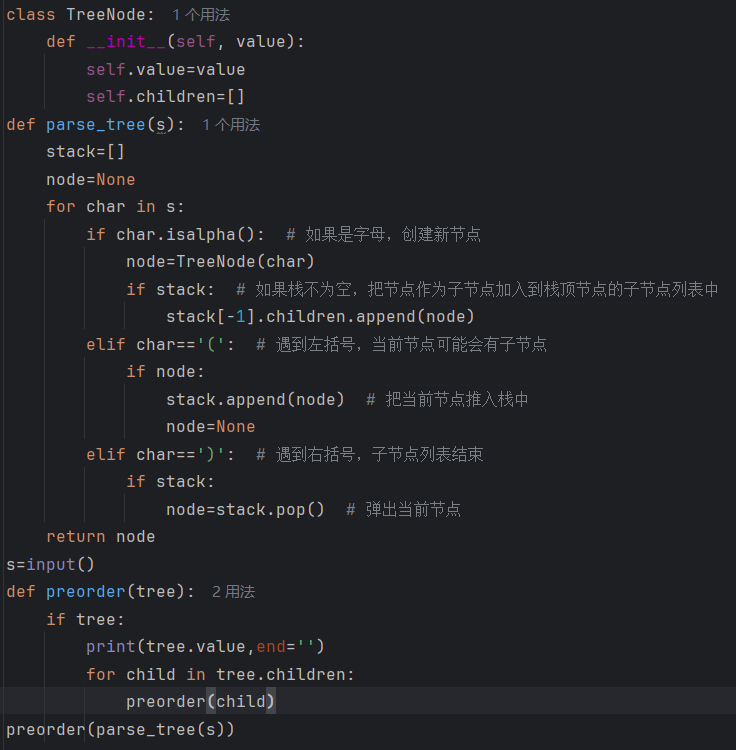
在循环过程中，index从数字开始取（依次+1），x依次取列表的元素



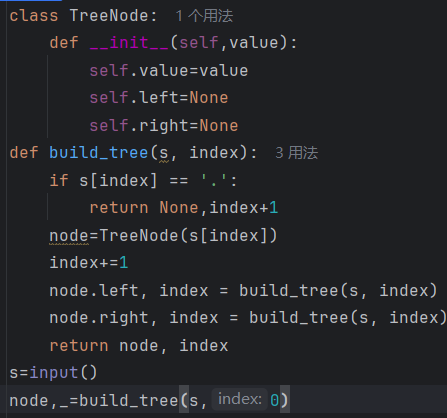
21. **括号嵌套二叉树Eg.s=A(B(\*,C),D(E,\*))**



**22.括号嵌套树Eg.** **A(B(E),C(F,G),D(H(I)))**

****

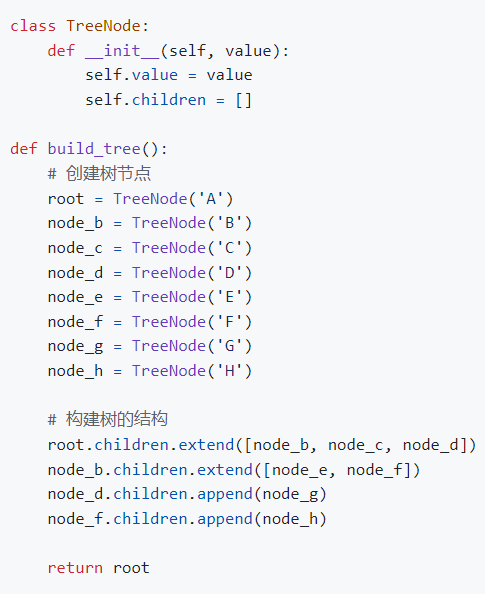
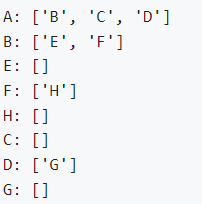
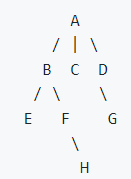
**23.扩展二叉树**

****

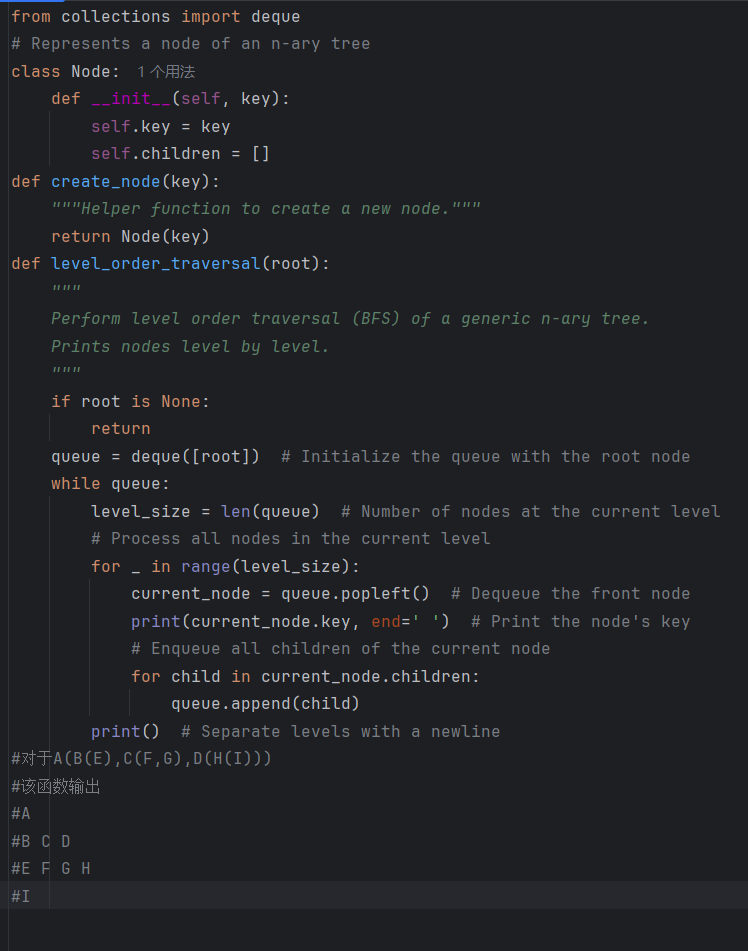
**Eg.s=ABD..EF..G..C..**

**24.邻接表构建树**

**Eg:**



**25.层序遍历**

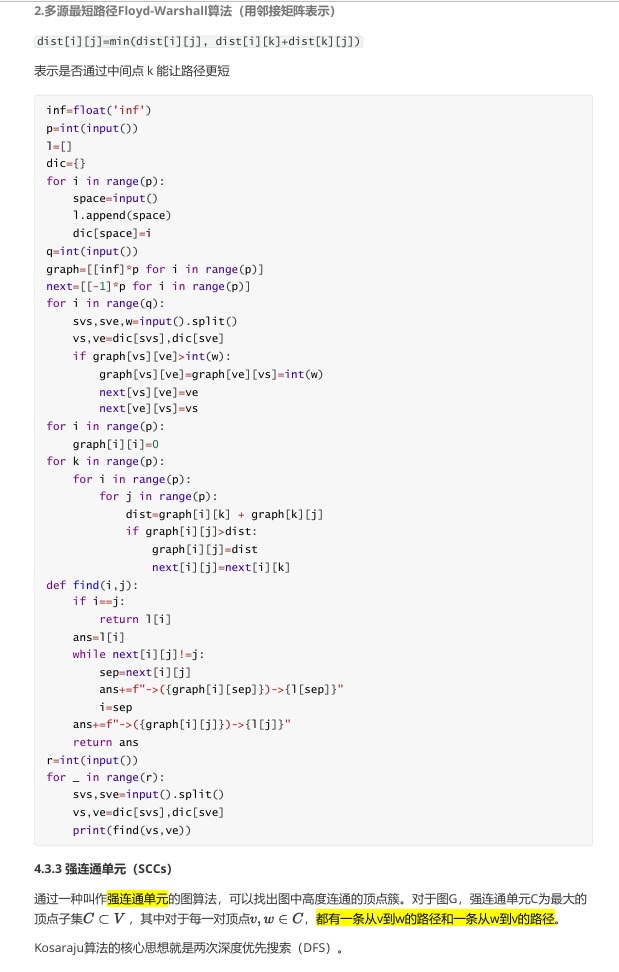
****

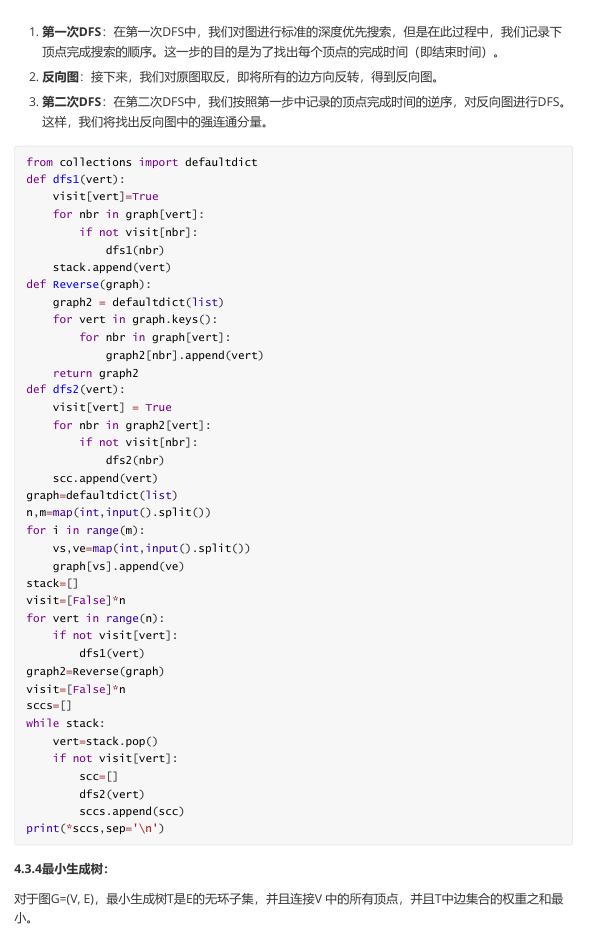
****

****

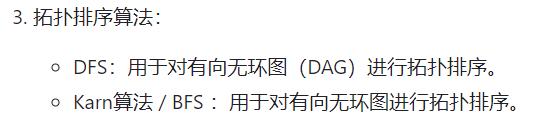
****

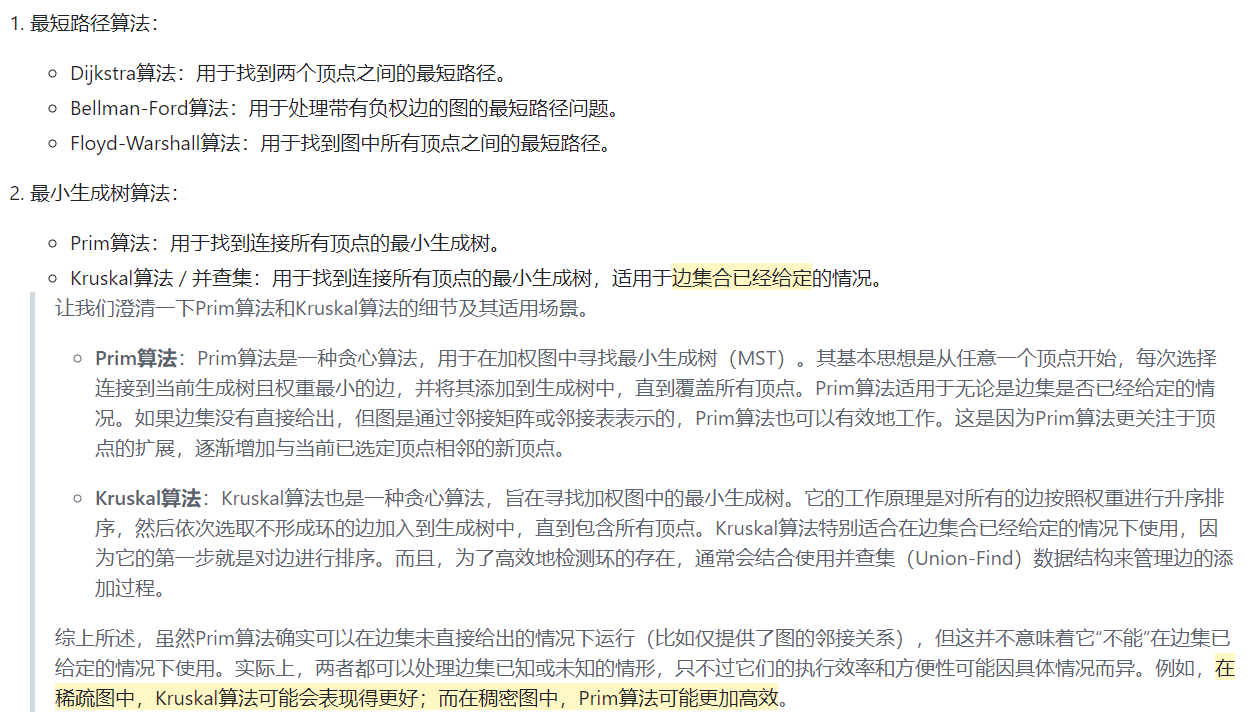
****

****

****

****

****



**拓扑排序：拓扑排序（Topological Sorting）是对有向无环图（DAG）进行排序的一种算法。它将图中的顶点按照一种线性顺序进行排列，使得对于任意的有向边 (u, v)，顶点 u 在排序中出现在顶点 v 的前面。**