1. **什么是图搜索过程？其中，重排Open表意味着什么？重排的原则是什么**？

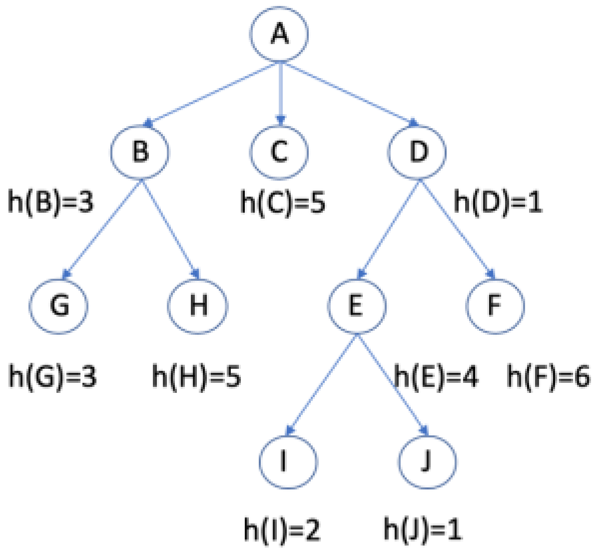
搜索算法是利用计算机的高性能来有目的的穷举一个问题解空间的部分或者所有的可能情况，从而求出问题的解的一种，图搜索过程就是来枚举解的可能情况来求出问题解的过程。其步骤一般如下：

1. 初始化，将初始结点放进open表，将close表置为空
2. Loop：if open = ()， Exit（Fail）
3. 取出open表的第一个结点n, 然后将其放进close表
4. 如果结点n为目标结点，Exit(Success)
5. Expand(n) -> {mi}, 计算耗散值
6. 根据相应的规则来判断扩展出的结点不在open表和close表，在open表，在close表的情况如何处理
7. 对open表中的结点排序
8. Go Loop

重排open表即意味着每次图搜索的循环里优先扩展哪个结点(即决定第3步取出的结点的优先次序)，而不同的搜索对应不同的重排规则。

重排的原则对应于需要的不同搜索策略(如A*算法想尽快找到最优解，若h单调时可以取f从小到大的次序)

2.试给出爬山法和分支界限搜索算法搜索图1 所示的从A到J的搜索路径，其中g(n)用节点深度表示，h(n)的值在图中显示。



爬山法：

一开始open表里的元素只有A

每次取出元素	对应新的open表（只根据h排序）
A	D(1), B(3), C(5)
D	B(3), E(4), C(5), F(6)
B	G(3), E(4), C(5), H(5), F(6)
G	E(4), C(5), H(5), F(6)
E	J(1), I(2), C(5), H(5), F(6)
J	

分支界限法：

一开始open表里的元素只有A

每次取出元素	对应的open表(只根据g排序)
A	B(1), C(1), D(1)
B	C(1), D(1), G(2), H(2)
C	D(1), G(2), H(2)
D	G(2), H(2), E(2), F(2)
G	H(2), E(2), F(2)
H	E(2), F(2)
E	F(2), I(3), J(3)
F	I(3), J(3)
I	J(3)
J	

3. **怎么用一架天平3次称出13个硬币中唯一的然而未知轻重的假币（已知有标准的硬币）**

   首先对于1个小问题，若是3枚硬币有1枚假的是轻，如何一次鉴别出：

   任取两枚硬币放在天平两侧，若是左边轻，则左边那枚是假，若是右边轻，则右边那边是假，若是平，则剩下那枚是假。

   同理对于3枚硬币有1枚是重也可以如上一次判断出。

   然后对于13枚硬币：

   ①首先第一次在天平两边各放4个，剩下5个

   ②若是天平两边不平，说明假币在这8个中，跳转③， 若是天平两边平，说明假币在那5个中，跳转④

   ③8个假币4个在轻组，4个在重组，取3轻1重和1轻3标准在两侧

   若是3轻1重的这边重，则说明是这边的重组那枚重或者另一边轻组那枚的轻，即缩小在2枚中，用一次标准硬币得到答案；

若是3轻1重的这边轻，则说明这3枚硬币里有1枚轻，则归结为上面的小问题一次解决

若是两边一样重，则说明是剩下的3枚硬币里有一枚重，则归结为上面的小问题一次解决

此时问题已经解决，跳出搜索过程

④拿3个标准的硬币(可以是剩下的8个中)与5个中的3个测：

若是平衡的话，则说明假币在剩2个中，拿一个标准的与2个中的1个测，若是平，则假币是另一个，若是不平，则假币是测的那一个；

若测的3个重，则说明假币在这3个中且假币重，则归结为上面的小问题一次解决

若测的3个轻，则说明假币在这3个中且假币轻，也归结为上面的小问题一次解决

此时问题已经解决，跳出搜索过程

4.给定4升和3升的水壶各一个。水壶上没有刻度。可以向水壶中加水。如何在4升的壶中准确的得到2升水?

可以定义一开始的状态为(0, 0)，first表示4升水壶的状态，second表示3升水壶的状态，然后每个状态有6种扩展方式： 将4升水壶倒满，将4升水壶倒空，将3升水壶倒满，将3升水壶倒空，将4升水壶的水倒入3升水壶(直到4升为空或者3升满)，将3升水壶的水倒入4升水壶(直到3升为空或者4升满)。

以上就可以通过一个图搜索过程来实现。

通过模拟该过程，得到一组可行解：

①将4升水壶装满（4， 0）

②4升水壶倒3升倒满 （1， 3）

③将3升水壶倒空 （1， 0）

④4升水壶倒3升倒完 （0， 1）

⑤4升水壶倒满 （4， 1）

⑥4升水壶倒3升倒满 （2， 3） Exit(Success)

5.对于A* 算法，证明下面的结论：

a) 对于有限图，A* 算法一定会在有限步内终止；

b) 对于无限图，只要从初始节点到目标节点有路径存在，则A* 算法也必然会终止；

若存在路径，则A* 算法一定会终止在最优路径上。

a)对于一个有限图，从初始节点到某个结点要么没有路径，要么肯定存在最短路，若是没有路径，则不会被扩展更新，即这个结点不会增加步数，而若是存在最短路，必然在更新有限次后到达最短路（从初始结点到这个结点的路径个数是有限的），所以每个结点最多只有有限次个贡献，总共有有限次个结点，则步数一定是有限的。

b). (1)①首先对于一个无限图，若是**A***不结束时，open表中即使最小的1个f值也将增大到任意大, 因为g的值在不断增大，每次都有增量，无限增大则必然会到无穷大，即对open表中任意的n，使得 **f(n) > f*(s)**

②而**A***算法结束前，必然存在n, 使得**f(n) <= f*(s)**

所以若是不能有限步结束，则上述两条陈述互相矛盾，所以A*算法必然会在有限步结束。

(2) 若是最后扩展的结果不是最佳解，记目标为t, 即$f(t) > g(t) = f^*(s)$

而由②可得结束前open表中存在节点n，是的**$f(n) <= f^*(s)$**, 所以**$f(n) < f(t)$**,所以此时open表不会取t扩展而是取n扩展，即发生矛盾，假设不真

所以若存在路径，则A* 算法一定会终止在最优路径上。