

作业一

1. 考虑一个实时的在线电话翻译系统，该系统实现英语与日语之间的实时在线翻译，讨论该系统的性能度量，环境，执行器，感知器，并对该环境的属性进行分析。**10 每点 2 分**

【Answer】

性能度量：

翻译的正确率 **(2 分)**

环境：电话线路 **(2 分)**

传感器：麦克风 **(2 分)**

执行器：音响 **(2 分)**

完全可观察的，单 agent，确定的(无噪音条件下)，片段的，静态的，离散的。**(2 分)**

2. 考虑一个医疗诊断系统的 agent，讨论该 agent 最合适的种类(简单 agent,基于模型的 agent,基于目标的 agent 和基于效用的 agent)并解释你的结论。**共 10 分**

【Answer】

utility-based agent 基于效用的 Agent **(2 分)**

能够治愈病人的方法有很多种，系统必须衡量最优的方法来推荐给病人 **8 分**

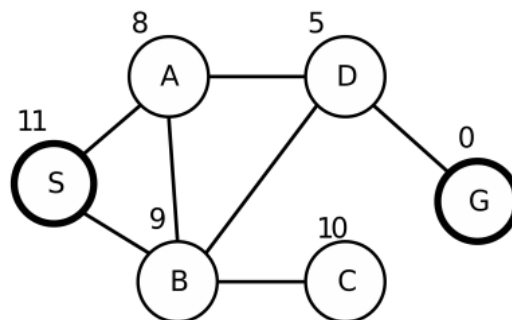
3. 先建立一个完整的搜索树，起点是 S,终点是 G,如下图,节点旁的数字表示到达目标状态的距离，然后用以下方法表示如何进行搜索，并分析几种算法的完备性、最优性、以及时间复杂度和空间复杂度。**40，每题 10 分 图 2 分 完备性、最优性、时间与空间复杂度分别为 2 分**

(a).深度优先；

(b).宽度优先；

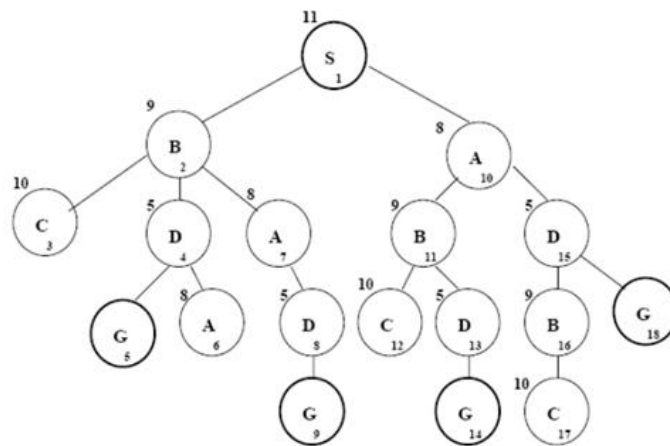
(c).爬山法；

(d).最佳优先；

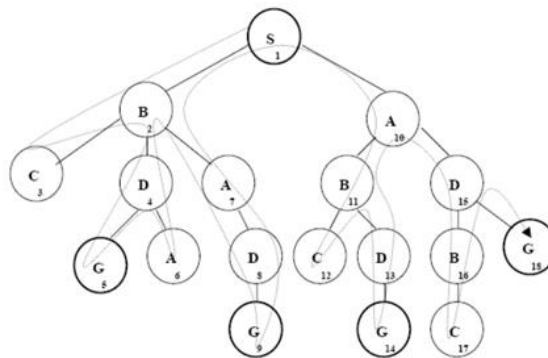


图一

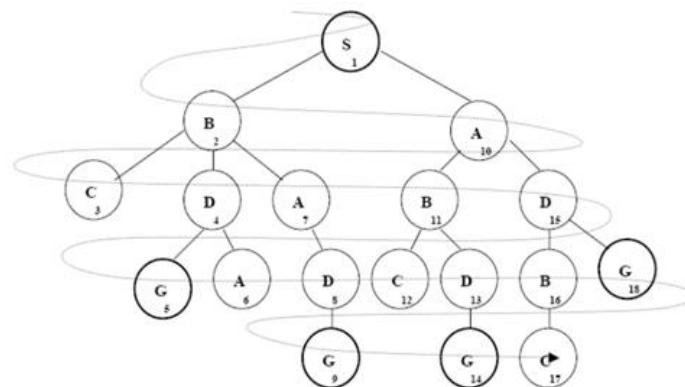
【Answer】：建树：



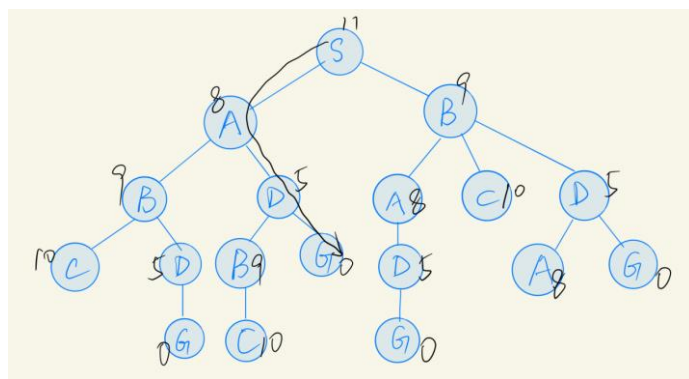
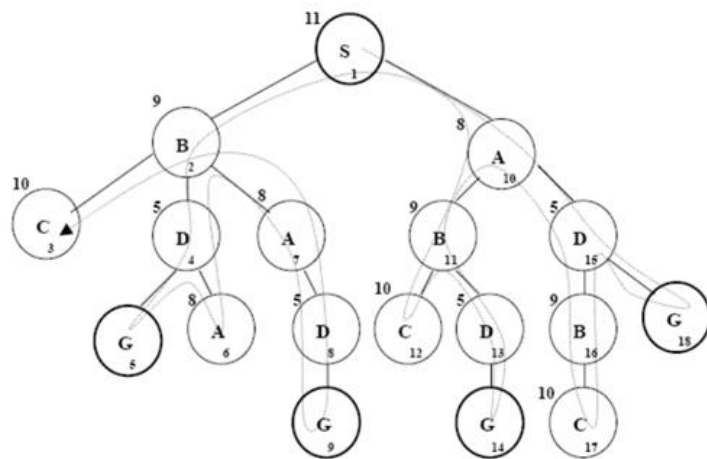
深度：



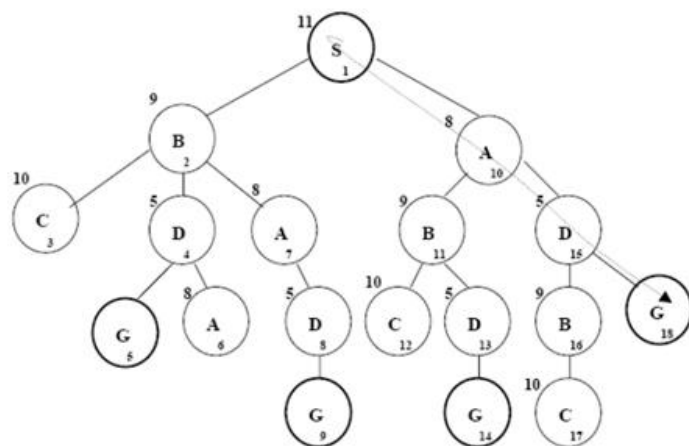
宽度：



爬山法：



最佳优先搜索：



以上几种算法的完备性、最优性、时间复杂度和空间复杂度分析如下：

深度优先搜索：在有限状态空间中，深度优先搜索是完备的，因为它至多扩展所有结点，但在树搜索中，则不完备（因为算法可能会陷入死循环）；该算法无法避免冗余路径，因此不是最优的；深度优先搜索只需要存储 $O(bm)$ 个结点，因此空间复杂度为 $O(bm)$ ；该算法可能搜索树上每个结点，因此时间复杂度为 $O(b^m)$ 。（其中 b 为分支因子， m 为树的最大深度）

广度优先搜索：首先，广度优先搜索是完备的（如果最浅的目标结点处于一个有限深度 d ，广度优先搜索在扩展完比它浅的所有结点之后最终一定能找到该目标结点；其次，如果路径代价是基于结点深度的非递减函数，则广度优先搜索是最优的；该方法的时间复杂度为 $O(b^d)$ （假设解的深度为 d ）；该方法的空间复杂度同样为 $O(b^d)$ （因为有 $O(b^d)$ 个结点在边缘结点中）。

爬山法：

(c).爬山法：

考虑随机重启时，全程遍历，时间与空间复杂度可以为 $O(b^d)$

(1) 利用最陡爬山法

最陡爬山法是一个简单的循环过程，不断地向值增加的方向持续移动，这里选择的距离最小的值移动。爬山算法,是一种局部贪心的最优算法. 该算法的主要思想是:每次拿相邻点与当前点进行比对,取两者中较优者,作为爬坡的下一步。在 S 点出发，因为 $h(A)=8$ ，小于 $h(B)$ ，所以爬山爬到 A ，之后类似这样就到 D ,最后到 G 。路径就是 $S-A-D-G$ 。

最陡爬山法不具有完备性，因为会经常陷入困境，不一定能找到解。而且只能找到多个局部最优点的其中一个，不一定是全局最优，所以不具有最优性。因为从上到下，每层只生成和保存一个结点，所以时间复杂度和空间复杂度都是 $O(d)$ 。 d 为目标结点或者局部最优结点的深度。

(2) 首选爬山法

依次寻找该点 x 的邻近点中首次出现的比点 x 价值高的点,并将该点作为爬山的点。依次循环,直至该点的邻近点中不再有比其大的点。

利用首选爬山法可能得到以上 3 条可能的路径，分别用不同颜色标注。

在 S 的时候，随机生成 A ,因为 A 比当前较优，所以从 A 再随机生成下一个结点，因为 $B=9$ ，比当前差，放弃再生成 D ,最后得到 $S \rightarrow A \rightarrow D \rightarrow G$ 这条路径。另外两条也是根据这个方法推出。

首选爬山法不具有最优性和完备性。因为从上到下，每层只生成和保存一个结点，所以时间复杂度和空间复杂度都是 $O(d)$ 。 d 为目标结点或者局部最优结点的深度

最佳优先搜索：

完备性：贪婪最佳优先搜索是不完备的（有限状态空间的图搜索版本是完备的）。 A^* 搜索是完备的。

最优性：贪婪最佳优先搜索：每次扩展是局部最优的选择，可能不能达到全局最优，所以 不是最优的。 A^* 搜索：如果 $h(n)$ 是可采纳的，则树搜索版本是最优的。如果 $h(n)$ 是一致的，则图搜索版本是 最优的。

时间复杂度：

贪婪最佳优先搜索：与深度优先类似，时间复杂度为 $O(b^m)$ 。

A* 搜索： $O(b^d)$ ，其中 $d=h^*-h$ ，为到目标结点的实际代价。

空间复杂度：

贪婪最佳优先搜索： $O(b^m)$ 。

A* 搜索： $O(b^d)$ ，其中 $d=h^*-h$ 。

4. 图二是一棵部分展开的搜索树，其中树的边记录了对应的单步代价，叶子节点标注了到达目标结点的启发式函数的代价值，假定当前状态位于结点 A。用下列的搜索方法来计算下一步需要展开的叶子节点。注意必须要有完整的计算过程，同时必须对扩展该叶子节点之前的节点顺序进行记录：每个 4 分，共 20 分

(1) 贪婪最佳优先搜索

(2) 一致代价搜索

(3) A*树搜索

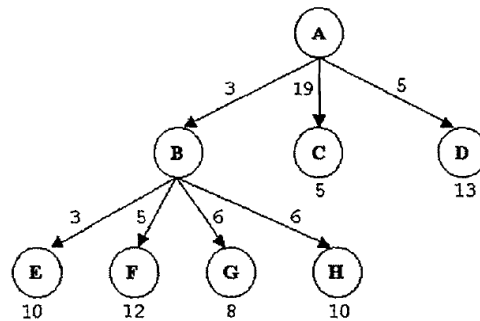
讨论以上三种算法的完备性和最优性。一致代价完备性 3 分，其余均 1 分

【Answer】：

贪婪最佳优先：如果 $h(B)>5$,首先访问叶子结点 C,如果 $h(B)\leq 5$,首先访问 B,再访问 C

一致代价搜索：B,D,E,F,G,H,C

A*树搜索：如果 $h(B)>15$,首先访问 D;如果 $h(B)\leq 15$,首先访问 B,再 E,G,D,H,F,C



图二

贪婪最佳优先搜索：该方法不能保证找到解，因此是不完备的；该方法不一定能找到最优解，因此不是最优的。

一致代价搜索：该方法在存在零代价行动时可能陷入死循环，因此是不完备的，如果每一步的代价都大于等于某个小的正值常数 ϵ ，那么一致搜索是完备的；一致代价搜索按照结点的最优路径扩展结点，因此是最优的。

A*树搜索：该方法能保证找到解，因此是完备的；因为 $h(n)$ 是可采纳的，因此 A*树搜索是最优的。

5. 给定一个启发式函数满足 $h(G)=0$, 其中 G 是目标状态, 证明如果 h 是一致的, 那么它是可采纳的。20

【Answer】:

假设 n 为任意一个状态, G 是任意一个目标状态。 $n, n_1, n_2, \dots, n_m, G$ 为从状态 n 到达状态 G 的一条最优路径, 我们已知

4 分 评估代价 $f(n)=g(n)+h(n)$

4 分 真实代价 $f'(n)=g(n)+c(n, a_1, n_1)+c(n_1, a_2, n_2)+\dots+c(n_m, a_{m+1}, G)$

4 分 目标: 证明 $f(n) \leq f'(n)$

8 分 证明:

$$\begin{aligned} f(n) &= g(n) + h(n) \leq g(n) + c(n, a_1, n_1) + h(n_1) \\ &\leq g(n) + c(n, a_1, n_1) + c(n_1, a_2, n_2) + h(n_2) \\ &\leq \dots \\ &\leq g(n) + c(n, a_1, n_1) + c(n_1, a_2, n_2) + \dots + c(n_m, a_{m+1}, G) + h(G) \\ &= f'(n) \end{aligned}$$