

第四章 作业

一、选择题（除已标注多选题，其他为单选题）

1. 在启发式搜索（有信息搜索）中，评价函数的作用是（ D ）
 - A. 计算从当前节点到目标节点之间的最小代价值
 - B. 判断搜索算法的空间复杂度
 - C. 判断搜索算法的时间复杂度
 - D. 从当前节点出发来选择后续节点
2. 在启发式搜索（有信息搜索）中，启发函数的作用是（ D ）
 - A. 从当前节点出发来选择后续节点
 - B. 判断搜索算法的时间复杂度
 - C. 判断搜索算法的空间复杂度
 - D. 计算从当前节点到目标节点之间的最小代价值
3. 在贪婪最佳优先搜索中，评价函数取值和启发函数取值之间的关系是（ D ）
 - A. 小于
 - B. 大于
 - C. 不相等
 - D. 相等
4. 在 A* 搜索算法中，评价函数可以如下定义（ A ）
 - A. 评价函数之值=(从起始节点出发到当前节点最小开销代价)+(从当前节点出发到目标结点最小开销代价)
 - B. 评价函数之值=(从起始节点出发到当前节点最小开销代价)/(从当前节点出发到目标结点最小开销代价)
 - C. 评价函数之值=(从起始节点出发到当前节点最小开销代价)*(从当前节点出发到目标结点最小开销代价)
 - D. 评价函数之值=(从起始节点出发到当前节点最小开销代价)-(从当前节点出发到目标结点最小开销代价)
5. A* 算法是一种有信息搜索算法，在最短路径搜索中引入的辅助信息是（ C ）
 - A. 路途中天气和交通状况等信息
 - B. 任意一个城市到起始城市之间直线距离
 - C. 任意一个城市到目标城市之间直线距离
 - D. 旅行者兴趣偏好信息
6. 为了保证 A* 算法是最优的，需要启发函数具有可容（admissible）和一致（consistency）的特点,下面对启发函数具有可容性这一特点的解释正确的是（ D ）
 - A. 启发函数是递增的
 - B. 启发函数是递减的
 - C. 启发函数不会过高估计从起始节点到目标结点之间的实际开销代价
 - D. 启发函数不会过高估计从当前节点到目标结点之间的实际开销代价

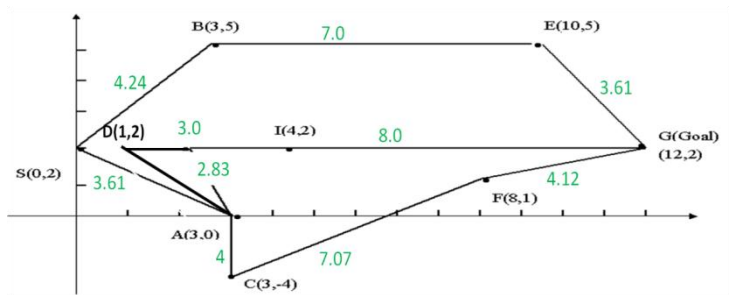
7. 有两种 8 皇后问题的形式化方式。“初始时 8 个皇后都放在棋盘上，然后再进行移动”是哪一种形式化方式？（ D ）
- A. 正则形式化
 - B. 完全形式化
 - C. 半态形式化
 - D. 增量形式化
8. 设{A, B, C, D}为变量，每个变量的域是{u, v, w}，且“!=”表示不等于，从如下表达式中选择那个是 CSP 形式化的 2 元约束？（ B ）
- A. $\langle A, A = v \rangle$
 - B. $\text{Diff}(A, D)$
 - C. $\text{Alldiff}(A, B, C, D)$
 - D. $A + B = C$
9. (多选)比较 CSP 和状态空间搜索，并从下列叙述中选择正确的答案。（ B D ）
- A. CSP 求解系统会比状态空间搜索求解系统慢。
 - B. CSP 可以快速排除大的搜索空间样本。
 - C. 状态空间搜索可以快速排除大的搜索空间样本。
 - D. CSP 求解系统会比状态空间搜索求解系统快。
10. 以下关于博弈的陈述哪个是正确的？（ B ）
- A. 经典搜索问题通常称为博弈。
 - B. 对抗搜索通常称为博弈。
 - C. 启发式搜索问题通常称为博弈。
 - D. 局部搜索问题通常称为博弈。
11. 下列智能体交互动作的总收益和损失可以小于或大于零的是哪种？（ C ）
- A. 计算机博弈
 - B. 双人博弈
 - C. 非零和博弈
 - D. 零和博弈
12. (多选)以下关于 $\alpha - \beta$ 剪枝的陈述哪些是正确的？（ B D ）
- A. $\alpha - \beta$ 剪枝旨在添加其搜索树中由 minimax 算法评价的大部分。
 - B. $\alpha - \beta$ 剪枝旨在消除其搜索树中由 minimax 算法评价的大部分。
 - C. $\alpha - \beta$ 剪枝旨在增加其搜索树中由 minimax 算法评价的节点数量。
 - D. $\alpha - \beta$ 剪枝旨在减少其搜索树中由 minimax 算法评价的节点数量。
13. 遗传算法是随机束搜索的一个变体，其中后继节点的生成是由：（ D ）
- A. 组合单一节点而不是组合两个双亲节点。
 - B. 组合单一状态而不是组合两个双亲状态。
 - C. 组合两个双亲节点而不是修改单一节点。
 - D. 组合两个双亲状态而不是修改单一状态。

11. (多选) 模拟退火算法的下列陈述哪些是正确的? (A C)
- A. 模拟退火算法的内循环与爬山法非常相似。
 - B. 模拟退火算法的内循环与爬山法完全不同。
 - C. 模拟退火算法不是选择最佳行动, 而是选择随机行动。
 - D. 模拟退火算法不是选择随机行动, 而是选择最佳行动。
12. (多选题) 禁忌搜索算法是一种: (A B C)
- A. 全局性邻域搜索算法
 - B. 模拟人类具有记忆功能的寻优特征
 - C. 通过禁忌准则来避免迂回搜索
 - D. 一旦进入禁忌表就不能被释放。
13. (多选题) 关于蚁群算法, 下面叙述正确的是 (A B D)
- A. 蚁群系统是一个增强型学习系统
 - B. 蚁群算法是通过人工模拟蚂蚁搜索食物的过程, 找寻蚁穴和食物之间的最短路径问题
 - C. 蚁群算法中, 蚂蚁选择路径是一种负反馈机制
 - D. 蚁群算法是一种应用于优化问题的启发式搜索算法
14. 粒子群算法用 (B) 表示问题的解。
- A. 粒子经历过的最优速度
 - B. 粒子经历过的最优位置
 - C. 粒子的速度
 - D. 粒子的位置
15. 遗传算法的基本操作顺序是 (A)
- A. 计算自适应度值、选择、交叉、变异
 - B. 计算自适应度值、交叉、选择、变异
 - C. 计算自适应度值、交叉、变异、选择
 - D. 选择、交叉、变异、计算自适应度值

二、简答题

1. 用爬山法求解下列问题:

已知: 如图所示, S 为初始点, G 为目标点。图中绿色标记数字为相邻结点间的实际距离。
各结点括号中的数值为坐标。各结点到目标 G 的直线距离公式如下,



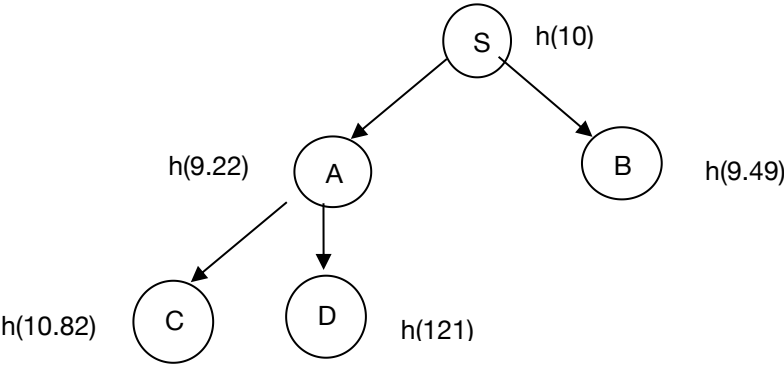
要求:

- (1) 给出问题的形式化描述。
- (2) 给出搜索树求解过程, 并记录生成节点数。

解:

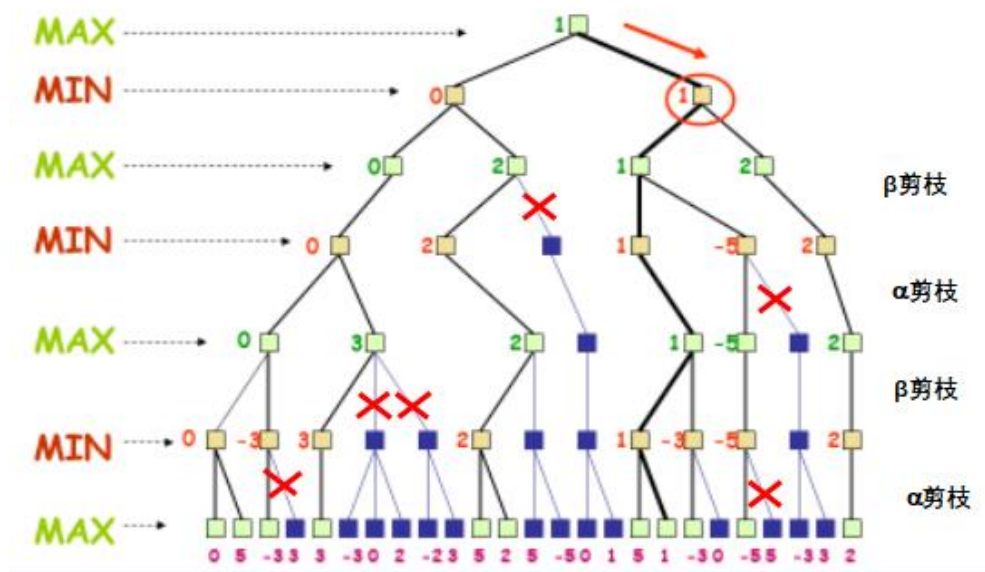
点	S	A	B	C	D	E	F	I	G	h
S (0,2)	0	3.61	4.24							10
A (3,0)		0		4	2.83					9.22
B (3,5)			0			7				9.49
C (3,-4)				0			7.07			10.82
D (1,2)					0			3		11
E (10,5)						0			3.61	3.61
F (8,1)							0		4.12	4.12
I (4,2)								0	8	8
G (12,2)									0	0

ClimbHill



Faille 返回 (失败返回)

2. 以 $\alpha - \beta$ 算法思想搜索，并标明何处发生何种剪枝？



三、CSP (约束可满足问题)

已知：4 个变量 A, B, C 和 D，每个变量有二个合法的取值。A 的取值为 A1, A2；B 的取值为 B1, B2；C 的取值为 C1, C2；D 的取值为 D1, D2。

每一对变量赋值需满足如下：

A-B: A1-B1, A2-B1

A-C: A1-C1, A2-C2

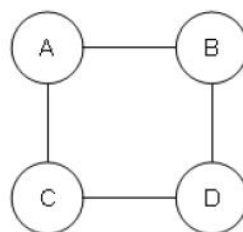
B-D: B1-D1

C-D: C2-D1

B-C: 没有约束.

A-D: 没有约束

其他均为非法取值。



Constraints:

A-B: A1-B1, A2-B1

A-C: A1-C1, A2-C2

B-D: B1-D1

C-D: C2-D1

A, B, C, D $\in \{1,2\}$

Figure 1: Formulate this problem as a CSP.

操作：每次为一个变量赋值，变量选取采用字母排序，取值按数值排序，直到产生有效解。

采用：(1) 回溯法（如表1）；(2) 带向前检测回溯法（如表2）求解。

试回答：

(a) 纯回溯法:在找到解之前进行多少次分派？

(b) 纯回溯法:给出解序列。

(c) 带向前检测回溯法:在找到解之前进行多少次分派？

(d) 带向前检测回溯法: 给出解序列。

表1 纯回溯法赋值

.Step 1	A1
Step 2	A1 B1
Step 3	A1 B1 C1
Step 4	A1 B1 C1 D1 ×
Step 5	A1 B1 C1 D2 ×
Step 6	A1 B1 C2 ×
.....	

表 2 带向前监测回溯法赋值

		A	B	C	D
Init		A1 A2	B1 B2	C1 C2	D1 D2
.Step 1	A1	A1	B1	C1	D1 D2
Step 2	A1 B1	A1	B1	C1	D1
Step 3	A1 B1 C1 ×	A1	B1	C1	
Step 4	A2	A2	B1	C2	D1 D2
.....					

解：（1）采用回溯法步骤如表1所示；

- （a）使用纯回溯按字母顺序检查这些变量并以数字顺序为这些变量分派情况如表1；
 （b）解为A2, B1, C2, D1。

step1	A1
step2	A1 B1
step3	A1 B1 C1
step4	A1 B1 C1 D1 ×
step5	A1 B1 C1 D2 ×
step6	A1 B1 C2 ×
step7	A1 B2 ×
step8	A2
step9	A2 B1
step10	A2 B1 C1 ×
step11	A2 B1 C2
step12	A2 B1 C2 D1 DONE!

Table 1: Assign values using pure backtracking strategy

（2）带向前检测回溯法步骤如表 2 所示；

- （c）使用回溯和正向检查可以发现比纯回溯更早的非法状态。变量分派情况如下：

(d) 解为A2, B1, C2, D1.

	A	B	C	D
init	1,2	1,2	1,2	1,2
step1 A1	1	1	1	1,2
step2 A1 B1	1	1	1	1
step3 A1 B1 C1 ×	1	1	1	
step4 A2	2	1	2	1,2
step5 A2 B1	2	1	2	1
step6 A2 B1 C2	2	1	2	
step7 A2 B1 C2 D1 DONE!	2	1	2	1

Table 2: Assign values using backtracking with forward checking