

1. Search – A* （4 分）

要点总结

- A* 使用

$f(n) = g(n) + h(n)$

- $g(n)$: 起点→n 的已知代价
- $h(n)$: 估计 n→goal 的代价 (heuristic)
- 使用 优先队列（open list）按 f 从小到大取节点；有 closed list 存已扩展节点。
- Admissible heuristic: $h(n) \leq h^*(n)$, 保证不高估 → A* 最优。
- Consistent heuristic: $h(n) \leq c(n, n') + h(n')$, 保证单调。
- 与 UCS/Greedy 的对比：
 - UCS: 只看 g
 - Greedy: 只看 h
 - A*: 看 g+h

3. Ensemble learning – Random forest （4 分）

要点总结

- Random Forest = Bagging + 随机特征子集：
 - bootstrap 采样形成多份训练集 → 每份训练一棵树
 - 每个 split 处随机选 k 个特征，从中选最佳分裂。
- 好处：
 - 降低 variance, 减少树间相关性, 抗过拟合。
- Out-of-Bag (OOB) error: 没被抽进某棵树训练集的样本，用来评估这棵树

5. Reinforcement learning – TD learning （4 分）

要点总结

- TD(0) 状态值更新:

$V(S_t) \leftarrow V(S_t) + \alpha[R_{t+1} + \gamma V(S_{t+1}) - V(S_t)]$

- 里面的括号是 TD error。
- SARSA (on-policy):

$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \alpha[R_{t+1} + \gamma Q(S_{t+1}, A_{t+1}) - Q(S_t, A_t)]$

- Q-learning (off-policy):

$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \alpha[R_{t+1} + \gamma \max_a Q(S_{t+1}, a) - Q(S_t, A_t)]$

7. Computer vision – Edge detection （2 分）

要点总结 （2 分会考很小）

- 目标: 检测强度变化大的位置 (边缘)。
- 方法：
 - 一阶: gradient (Sobel、Prewitt)
 - 二阶: Laplacian
 - Canny: Gaussian smoothing → gradient → non-maximum suppression → double threshold + hysteresis。

8. Computer vision – Averaging filter （4 分）

要点总结

- Averaging / mean filter：
 - 3×3 kernel 全是 1/9, 做卷积。
 - 用于去除噪声, 平滑图像, 但会模糊边缘。

- 对二值图像也可以用 sum > threshold 判 1, 否则 0。

10. Knowledge representation – Propositional logic （4

要点总结

- 命题逻辑符号: $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$ 。
- 真值表: 列出所有组合, 算表达式的 T/F。
- 常考推理形式: Modus Ponens 等。
- 可能涉及: 把英语描述转成逻辑式。

可能题型

- 把一段自然语言转为命题逻辑：
 - “如果在下雨并且我没带伞, 那我会淋湿。”
 $\rightarrow R \wedge \neg U \rightarrow W$

2. Decision Trees – Entropy （4 分）

要点总结

- Entropy:

$H(S) = - \sum_i p_i \log_2 p_i$

- 信息增益:

$Gain(S, A) = H(S) - \sum_v \frac{|S_v|}{|S|} H(S_v)$

- 直观：
 - 子集越“纯” (一个类占绝对多数) → entropy 低。
 - 完全 50/50 → entropy = 1 (对两类)。
- Patrons vs Type 例子：
 - Patrons 的加权 entropy=0.459
 - Type 的 entropy=1 → Patrons 更好。

4. Neural networks – Perceptron （4 分）

要点总结

- 单个感知机:

$s = w_0 + \sum_i w_i x_i, \quad y = step(s)$

- 可以实现 AND、OR、NOT 等线性可分函数; 不能实现 XOR。
- AND 示例:
 $w_0 = -1.5, w_1 = 1, w_2 = 1 \rightarrow$ 同时为 1 才输出 1。
- 多层感知机可以表示 XOR 和任意布尔函数。

6. Optimisation – Genetic algorithms （4 分）

要点总结

- Genotype 表示为串 (bit string / real-valued 等)。
- 一代包括：
 - Selection (轮盘赌 / tournament)
 - Crossover (单点/两点/均匀)
 - Mutation (按小概率翻转位)
- Fitness function 决定选择概率。
- GA 优点: 全局搜索、适合非凸问题; 缺点: 慢、不保证精确最优。

9. Language processing – Trigram probability （4 分）

要点总结

- n-gram 模型:

$P(w_1, \dots, w_T) \approx \prod_t P(w_t \mid w_{t-n+1}, \dots, w_{t-1})$

- trigram:

$P(w_i \mid w_{i-2}, w_{i-1}) \approx \frac{C(w_{i-2}, w_{i-1}, w_i)}{C(w_{i-2}, w_{i-1})}$

- 可能用 Laplace smoothing 或加伪计数。

11. Uncertainty – Bayes’ theorem （4 分）

要点总结

- Bayes 定理:

$P(H \mid E) = \frac{P(E \mid H)P(H)}{P(E)}$

- 总概率:

$P(E) = \sum_i P(E \mid H_i)P(H_i)$

- 典型例子: 医疗检测、假阳性/假阴性、spam detection。

要点总结

- 来自 Week 10 “Human-aligned intelligent robotics”
- Powell 提到的 AI Level（大意）：
 - 工具型 AI（Tool）
 - 助理型 AI
 - 协作型 AI
 - 人类对齐 / 自主智能体 等层级
- 你只需要记住：从“被动工具”到“主动协作、对齐人类价值”的发展维度。

可能题型

- 让你列出/简要描述 2–3 个 AI level 及各自特点。
- 问：为什么“human-aligned AI / robotics”是更高层次目标？（强调安全、可解释、人机协作）

12. Uncertainty – Fuzzy logic （6 分）

6 分说明这里题会稍大一点，可能包括计算 + 概念。

要点总结

- Fuzzy set：元素有 membership degree $\mu_A(x) \in [0, 1]$ 。
- 基本操作：
 - $\mu_{A \cap B}(x) = \min(\mu_A(x), \mu_B(x))$
 - $\mu_{A \cup B}(x) = \max(\mu_A(x), \mu_B(x))$
 - $\mu_{\neg A}(x) = 1 - \mu_A(x)$
- 常见 membership function：triangular, trapezoidal, Gaussian ...
- Fuzzy rules：“IF temperature is HIGH AND humidity is LOW THEN fan_speed is FAST”
- 模糊推理步骤：
 - fuzzification（输入模糊化）
 - rule evaluation
 - aggregation
 - defuzzification（如 centroid）

$$\nabla^2 f(x, y) \approx f(x + 1, y) + f(x - 1, y) + f(x, y + 1) + f(x, y - 1) - 4f(x, y)$$

➤ We can write the kernel as follow:

$$\begin{pmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{pmatrix}$$

方法	类型	检测效果
Sobel/Prewitt	一阶梯度	方向敏感 edge detection
Laplacian	二阶导	方向不敏感 zero-crossing edges

🔍 用表格看清楚：

实际情况	模型预测	名称	含义	📄
🔴 有病	🟢 预测有病	True Positive	正确预测有病	
🔴 有病	⚪ 预测没病	False Negative ❌	把应检测出来的病漏掉（漏检）→ 把对的判错	
⚪ 没病	⬛ 预测有病	False Positive ❌	把没病的人误判成有病（误报）→ 把错的判对	
⚪ 没病	⚪ 预测没病	True Negative	正确预测没病	