

实验一 产生式系统

一、实验目的：（时间：3月23日）

1. 熟悉知识的表示方法
2. 掌握产生式系统的运行机制
3. 理解产生式系统推理的基本方法

二、实验内容

运用所学知识，设计并编程实现一个小型动物识别系统，能识别虎、金钱豹、斑马、长颈鹿、鸵鸟、企鹅、信天翁等七种动物的产生式系统。

规则库：

- r1: IF 该动物有毛发 THEN 该动物是哺乳动物
- r2: IF 该动物有奶 THEN 该动物是哺乳动物
- r3: IF 该动物有羽毛 THEN 该动物是鸟
- r4: IF 该动物会飞 AND 会下蛋 THEN 该动物是鸟
- r5: IF 该动物吃肉 THEN 该动物是食肉动物
- r6: IF 该动物有犬齿 AND 有爪 AND 眼盯前方
THEN 该动物是食肉动物
- r7: IF 该动物是哺乳动物 AND 有蹄
THEN 该动物是有蹄类动物
- r8: IF 该动物是哺乳动物 AND 是反刍动物
THEN 该动物是有蹄类动物
- r9: IF 该动物是哺乳动物 AND 是食肉动物 AND 是黄褐色
AND 身上有暗斑点 THEN 该动物是金钱豹
- r10: IF 该动物是哺乳动物 AND 是食肉动物 AND 是黄褐色
AND 身上有黑色条纹 THEN 该动物是虎
- r11: IF 该动物是有蹄类动物 AND 有长脖子 AND 有长腿
AND 身上有暗斑点 THEN 该动物是长颈鹿
- r12: IF 该动物有蹄类动物 AND 身上有黑色条纹
THEN 该动物是斑马
- r13: IF 该动物是鸟 AND 有长脖子 AND 有长腿 AND 不会飞
AND 有黑白二色 THEN 该动物是鸵鸟
- r14: IF 该动物是鸟 AND 会游泳 AND 不会飞
AND 有黑白二色 THEN 该动物是企鹅

r15: IF 该动物是鸟 AND 善飞 THEN 该动物是信天翁

要求给定初始条件，能识别出是哪种动物。

比如已知初始事实存放在综合数据库中：

暗斑点，长脖子，长腿，奶，蹄

运行后得该动物是：长颈鹿

三、实验过程：

1. 应用所学知识，编写产生式系统实验程序，建立知识库，运行推理求解。

2. 产生式系统实验程序，构建规则库如下所示。系统分为两部分，第一部分是初始化综合数据库，分为两类，一类是间接数据库，即不指向最终动物的信息。另一类是直接数据库，即指向最终动物的信息。使用的是二维列表加上一个一维列表存储 key 与 value。

```
构建规则库：
构建间接数据库，输入0结束：
请输入条件，以空格隔开：有毛发
请输入结果：哺乳动物
请输入条件，以空格隔开：有奶
请输入结果：哺乳动物
请输入条件，以空格隔开：有羽毛
请输入结果：鸟
请输入条件，以空格隔开：会飞 会下蛋
请输入结果：鸟
请输入条件，以空格隔开：吃肉
请输入结果：食肉动物
请输入条件，以空格隔开：犬齿 有爪 眼盯前方
请输入结果：食肉动物
请输入条件，以空格隔开：哺乳动物 有蹄
请输入结果：有蹄类动物
请输入条件，以空格隔开：0
构建直接数据库，输入0结束：
请输入条件，以空格隔开：哺乳动物 食肉动物 黄褐色 暗斑点
请输入结果：金钱豹
请输入条件，以空格隔开：哺乳动物 食肉动物 黄褐色 黑色条纹
请输入结果：虎
请输入条件，以空格隔开：有蹄类动物 长脖子 长腿 暗斑点
请输入结果：长颈鹿
请输入条件，以空格隔开：有蹄类动物 黑色条纹
请输入结果：斑马
请输入条件，以空格隔开：鸟 长脖子 长腿 不会飞 黑白二色
请输入结果：鸵鸟
请输入条件，以空格隔开：鸟 会游泳 不会飞 黑白二色
请输入结果：企鹅
请输入条件，以空格隔开：0
初始化完毕！
```

第二部分为添加数据库和识别动物，查询的时候添加，可以有效的扩增数据库，使功能更灵活与完善。在识别动物的时候，先在间接数据库中完善信息，在直接数据库中查找对应的动物，记录信息时，使用字典记录某个特征是否出现，这样查询的复杂度就为 $O(1)$

```
请输入接下来操作(1. 添加间接数据库 2. 添加直接数据库 3. 查询动物 0. 结束操作): 1
请输入条件, 以空格隔开: 哺乳动物 反当动物
请输入结果: 有蹄类动物
请输入接下来操作(1. 添加间接数据库 2. 添加直接数据库 3. 查询动物 0. 结束操作): 2
请输入条件, 以空格隔开: 鸟 善飞
请输入结果: 信天翁
请输入接下来操作(1. 添加间接数据库 2. 添加直接数据库 3. 查询动物 0. 结束操作): 3
请输入动物主要特征(使用空格分开): 鸟 善飞
该动物: 信天翁
请输入接下来操作(1. 添加间接数据库 2. 添加直接数据库 3. 查询动物 0. 结束操作): 3
请输入动物主要特征(使用空格分开): 暗斑点 长脖子 长腿 有奶 有蹄
该动物: 长颈鹿
请输入接下来操作(1. 添加间接数据库 2. 添加直接数据库 3. 查询动物 0. 结束操作): 3
请输入动物主要特征(使用空格分开): 有毛发 吃肉 黄褐色 暗斑点
该动物: 金钱豹
请输入接下来操作(1. 添加间接数据库 2. 添加直接数据库 3. 查询动物 0. 结束操作):
```

图 1 产生式系统实验程序界面

3. 在理解产生式规则作为知识表示和产生式系统推理的基础上, 提交实验报告。

实验二 过河问题

一、实验目的（时间：3月23日）

1. 理解并掌握深度优先搜索算法
2. 理解递归的思想

二、实验内容

有 N 个传教士和 N 个野人来到河边渡河，河岸有一条船，每次至多可供 k 人乘渡。问：传教士为了安全起见，应如何规划摆渡方案，使得任何时刻，河两岸以及船上的野人数目总是不超过传教士的数目(否则不安全，传教士有可能被野人吃掉)。即求解传教士和野人从左岸全部摆渡到右岸的过程中，任何时刻满足 M (传教士数) $\geq C$ (野人数) 和 $M+C \leq k$ 的摆渡方案。

三、实验过程

1. 算法分析

假设以传教士和野人的数量 N 都为 3，船一次最大的载人量 K 为 2 人。

初始状态：河左岸有 3 个野人、3 个传教士；河右岸有 0 个野人和 0 个传教士；船停在左岸，船上有 0 个人。初始状态 (3, 3, 1)

目标状态：河左岸有 0 个野人和 0 个传教士；河右岸有 3 个野人和 3 个传教士；船停在右岸，船上有 0 个人。目标状态 (0, 0, 0)

将整个问题抽象成怎样从初始状态经一系列的中间状态从而达到目标状态，状态的变化是通过划船渡河来引发的。

以河的左岸为基点来考虑

P_{ij} ：把船从左岸划向右岸，其中，第一下标 i 表示船载的传教士数，第二下标 j 表示船载的食人者数；

Q_{ij} ：从右岸将船划回左岸，下标的定义同前。

则共有 10 种操作，操作集为 $F = \{P_{01}, P_{10}, P_{11}, P_{02}, P_{20}, Q_{01}, Q_{10}, Q_{11}, Q_{02}, Q_{20}\}$

规则集合

P_{10} if ($ML, CL, BL=1$) then ($ML-1, CL, BL-1$)

P_{01} if ($ML, CL, BL=1$) then ($ML, CL-1, BL-1$)

P_{11} if ($ML, CL, BL=1$) then ($ML-1, CL-1, BL-1$)

P_{20} if ($ML, CL, BL=1$) then ($ML-2, CL, BL-1$)

P_{02} if ($ML, CL, BL=1$) then ($ML, CL-2, BL-1$)

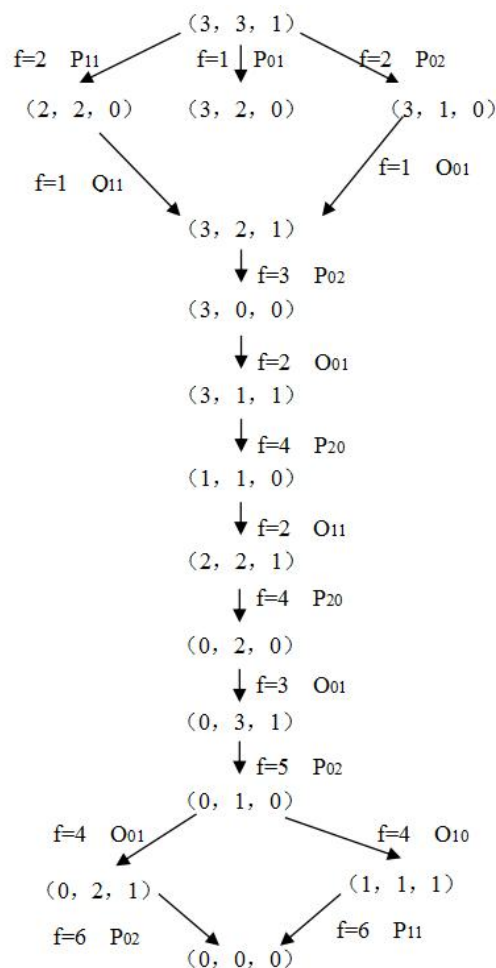
Q10 if (ML ,CL , BL=0) then (ML+1 , CL , BL+1)
 Q01 if (ML ,CL , BL=0) then (ML , CL+1 , BL +1)
 Q11 if (ML ,CL , BL=0) then (ML+1 , CL +1, BL +1)
 Q20 if (ML ,CL , BL=0) then (ML+2 , CL +2, BL +1)
 Q02 if (ML ,CL , BL=0) then (ML , CL +2, BL +1)

根据要求，共得出以下 5 中可能的渡河方案：

- (1) 渡 2 传教士
- (2) 渡 2 野人
- (3) 渡 1 野人 1 传教士
- (4) 渡 1 传教士
- (5) 渡 1 野人

本程序要定义状态结点，使用集合存储状态结点，使用递归的思想来寻找目标状态。

2.程序执行流程



上述原理完全是按照传教士和野人数量为 3 人，船一次的载客量最大为 2 为前提所得。结合上述分析不难看出，当尚未规定传教士和野人具体数量以及船一次最大载客量为多少时，与之前分析的差别仅仅在于根据人数不同，从河一端到另一端一共有多少中 载客方式，

其他的原理流程与传教士和野人数量为 3 人，船一次的载客量最大为 2 的流程全部相同。

全部的可能状态共有 32 个，如表 1 所示。

表 1 传教士和食人者问题的全部可能状态

状 态	m, c, b	状 态	m, c, b	状 态	m, c, b	状 态	m, c, b
S ₀	3,3,1	S ₈	1,3,1	S ₁₆	3,3,0	S ₂₄	1,3,0
S ₁	3,2,1	S ₉	1,2,1	S ₁₇	3,2,0	S ₂₅	1,2,0
S ₂	3,1,1	S ₁₀	1,1,1	S ₁₈	3,1,0	S ₂₆	1,1,0
S ₃	3,0,1	S ₁₁	1,0,1	S ₁₉	3,0,0	S ₂₇	1,0,0
S ₄	2,3,1	S ₁₂	0,3,1	S ₂₀	2,3,0	S ₂₈	0,3,0
S ₅	2,2,1	S ₁₃	0,2,1	S ₂₁	2,2,0	S ₂₉	0,2,0
S ₆	2,1,1	S ₁₄	0,1,1	S ₂₂	2,1,0	S ₃₀	0,1,0
S ₇	2,0,1	S ₁₅	0,0,1	S ₂₃	2,0,0	S ₃₁	0,0,0

值得注意的是按照题目规定的条件，我们应该划去不合法的状态，这样可以加快搜索求解的效率。例如，首先可以划去岸边食人者数目超过传教士的情况，即 S₄、S₈、S₉、S₂₀、S₂₄、S₂₅ 等 6 种状态是不合法的；其次，应该划去右岸边食人者数目超过修道士的情况，即 S₆、S₇、S₁₁、S₂₂、S₂₃、S₂₇ 等情况；余下 20 种合法状态中，又有 4 种是不可能出现的状态：S₁₅ 和 S₁₆ 不可能出现，因为船不可能停靠在无人的岸边；S₃ 不可能出现，因为传教士不可能在数量占优势的食人者眼皮底下把船安全地划回来；还应该划去 S₂₈，因为传教士也不可能在数量占优势的食人者眼皮底下把船安全地划向对岸。可见，在状态空间中，真正符合题目规定条件的只有 16 个合理状态。

3.编写程序对问题进行求解

4.撰写实验报告

实验三 A*算法

一、实验目的：（时间：3月30日）

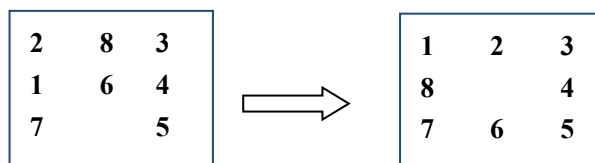
- 1.熟悉启发式搜索的定义、估价函数和算法过程
- 2.利用 A*算法求解 N 数码难题，理解求解流程和搜索顺序
- 3.熟练掌握 numpy 库的相关函数

二、实验原理：

A*算法是一种启发式图搜索算法，其特点在于对估价函数的定义上。对于一般的启发式图搜索，总是选择估价函数 f 值最小的节点作为扩展节点。因此， f 是根据需要找到一条最小代价路径的观点来估算节点的，所以，可考虑每个节点 n 的估价函数值为两个分量：从起始节点到节点 n 的实际代价以及从节点 n 到达目标节点的估价代价。

三、实验内容：

- 1、以 8 数码问题为例实现 A*算法的求解程序。



估价函数 $f(n) = g(n) + h(n)$

$g(n)=d(n)$ ——结点 n 在搜索树中的深度

$h(n)$ 可选择 $h1(n)$ ——结点 n 中“不在位”的数码个数 或 $h2(n)=p(n)=$ 将牌“不在位”的距离和（哈密尔顿距离或曼哈顿距离）

- 2、有能力的同学可以在求解 8 数码问题的 A*算法程序中，设置不同的初始状态和目标状态，针对不同的估价函数，求得问题的解，并比较它们对搜索算法性能的影响，包括扩展节点数、生成节点数等。
- 3、运行智能算法搜索教学软件中的八数码问题，加深对八数码问题求解流程的理解。
- 4、撰写实验报告，并画出 A*算法求解框图。

实验四 遗传算法应用

一、实验目的：（时间：3 月 30 日）

1. 熟悉和掌握遗传算法的原理、流程和编码策略
2. 利用遗传求解函数优化问题
3. 理解求解流程并测试主要参数对结果的影响。

二、实验原理：

遗传算法的基本思想正是基于模仿生物界遗传学的遗传过程。它把问题的参数用基因代表，把问题的解用染色体代表（在计算机里用二进制码表示），从而得到一个由具有不同染色体的个体组成的群体。这个群体在问题特定的环境里生存竞争，适者有最好的机会生存和产生后代。后代随机化地继承了父代的最好特征，并也在生存环境的控制支配下继续这一过程。群体的染色体都将逐渐适应环境，不断进化，最后收敛到一组最适应环境的类似个体，即得到问题最优的解。

四、实验内容：

1. 用遗传算法求解下列函数的最大值，设定求解精度到小数点后 4 位。

$$y = f(x_1, x_2) = \frac{1}{x_1^2 + x_2^2 + 1}$$

其中 $-5 \leq x_1, x_2 \leq 5$, 用遗传算法求解 y 的最大值。

- 1) 精度要求小数点后 4 位数字，采用二进制编码。种群规模 6，适应度函数为函数 $y=f(x_1, x_2)$ ，采用轮盘赌选择，采用单点交叉、单点变异，交叉概率 0.8，变异概率 0.1，迭代 100 次终止。
- 2) 画出遗传算法求解该问题的流程图
- 3) 编写程序，实现函数优化。