3-1 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？

答：

图搜索过程如下：

1. 建立一个只含有起始节点S的搜索图G，把S放到一个叫做OPEN的未拓展节点表中。
2. 建立一个叫做CLOSED的已拓展节点表，初始化为空表。
3. LOOP：若OPEN表是空表，则失败退出。
4. 选择OPEN表上的第一个节点，把它从OPEN表移出并放进CLOSED表中。称此节点为节点n。
5. 若n为一目标节点，则有解并成功退出，此解是追踪图G中沿着指针从n到S这条路径而得到的。
6. 扩展节点n，生成后继节点集合M。
7. 对那些未曾在G中出现过的（既未曾在OPEN表上，也未在CLOSED表上出现过的）M成员设置 其父节点指针指向n并加入OPEN表。对已经在OPEN或CLOSED表中出现过的每一个M成员，确定是否需要将其原来的父节点改为n。对已在CLOSED表上的每个M成员，若修改了其父节点，则将该节点从CLOSED表中移出，重新加入OPEN表中。
8. 按某一任意方式或按某个探试值，重排OPEN表。
9. GO LOOP。

重排OPEN表意味着，选出一个“最好”的节点在第6步进行扩展，不同的排序方式也会使图搜索过程的搜索策略不一样。

重排的原则应当视具体需求而定，不同的原则对应不同的搜索策略。一般地，重排的原则是：根据相应的图搜索策略，将“最好”的节点（即最有可能达到目标节点的节点）放在最前面。

* 1. 化为子句形有哪些步骤？请结合例子说明。

答：

1. **消去蕴涵符号**

将蕴涵符号转换为∨和~符号，如~A∨B替换A→B。

1. **减少否定符号的辖域**

每个否定符号~最多只用到一个谓词符号上，并反复应用狄摩根定律。如用~A∨~B代替~（A∧B），用。

1. **对变量标准化**

改名哑元（受量词约束的变元），保证每个量词用唯一的哑元，即不同量词约束的变元有不同的名字。如对 标准化得 。

1. **消去存在量词**

对全程量词辖域内的存在量词，以Skolem函数代替存在量词内的约束变量。对自由的存在量词，以一个新常量替代。如 代替 ，其中g(y)为Skolem函数。又如 代替 ，其中A为不含变量的Skolem函数即常量。

1. **化为前束形**

把所有全称量词移到公式的左边，并使每个量词的辖域包括这个量词后面公式的整个部分。

前束形 = {前缀} {母式}

（全称量词串）（无量词公式）

1. **把母式化为合取范式**

任何母式都可写成由一些谓词公式和谓词公式的否定的析取（子句）的有限集组成的合取。

如 代替 。

1. **消去全称量词**

余下的量词均被全称量词量化了。同时全称量词的次序也不重要。因此，可以消去前缀，即消去明显出现的全称量词。

1. **消去连词符号∧**

用{A, B}代替(A∧B)，消去符号∧。最后得到一个有限子句集，其中每个公式（子句）是文字的析取。

1. **更换变量名称**

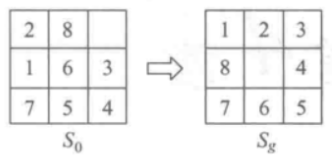
更换变量符号，使一个变量符号只出现在一个子句中。如P(x)∨Q(x)和~P(x)∨P(y)，更换变量符号后为P(x1)∨Q(x1)和~P(x2)∨P(y)。

* 1. 如何通过消解反演求取问题的答案？

答：

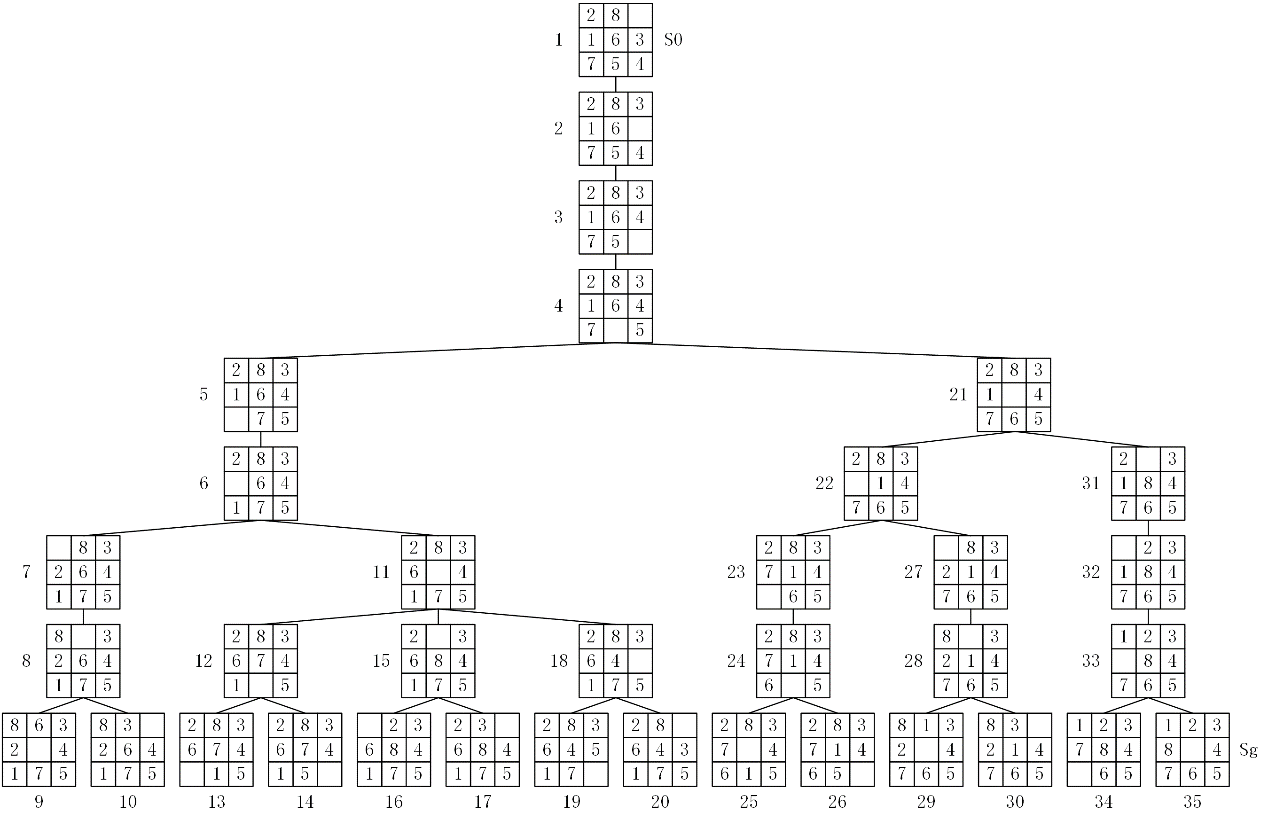
1. 给出公式集{S}和目标公式L；
2. 否定L，得到~L；
3. 把~L添加到S中；
4. 把新产生的集合{~L，S}化成子句集；
5. 应用消解原理，推导出一个表示矛盾的空子句。

3-7 用有界深度优先搜索方法求解八数码难题。



答：

设置深度界限为9，其搜索图如下所示：



按顺时针方向，最大深度为5。

3-11 规则演绎系统和产生式系统有哪几种推理方式？各自的特点是什么？

规则演绎系统的推理方式有正向推理、逆向推理和双向推理。

|  |  |  |
| --- | --- | --- |
|  | **正向推理** | **逆向推理** |
| **事实表达式** | 任意形 | 合取形 |
| **规则形式** | 单文字L→W | W→单文字L |
| **目标公式** | 文字析取 | 任意形 |
| **推理方向** | 从事实出发，正向应用规则，以目标为结束的一致解图 | 从目标出发，逆向应用规则，以事实为结束的一致解图 |

双向推理组合了正向推理和逆向推理的优点，具有更高的搜索求解效率。

产生式系统的推理方式有正向推理、逆向推理和双向推理。

|  |  |  |
| --- | --- | --- |
|  | **正向推理** | **逆向推理** |
| **驱动方式** | 数据驱动 | 目标驱动 |
| **推理方式** | 从一组数据出发先前推导结论 | 从可能的解答出发，向后推理验证解答 |
| **启动方式** | 从一个事件启动 | 由询问关于目标状态的一个问题而启动 |
| **透明程度** | 不能解释其推理过程 | 可解释其推理过程 |
| **推理方向** | 由底向上推理 | 由顶向下推理 |
| **典型系统** | CLIPS，OPS | PROLOG |

双向推理综合了正向推理和逆向推理的长处，克服了两者的短处，其控制策略比另外两种都要复杂。

3-14 下列语句是一些几何定理，把这些语句表示为基于规则的几何证明系统的产生式规则

1. 两个全等三角形的各对应角相等。
2. 两个全等三角形的各对应边相等。
3. 各对应边相等的三角形是全等三角形。
4. 等腰三角形的两底角相等。

答：

规则1： IF 两个三角形全等

THEN 各对应角相等

规则2： IF 两个三角形全等

THEN 各对应边相等

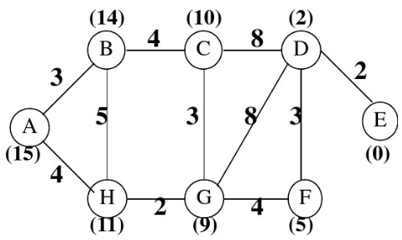
规则3： IF 两个三角形各对应边相等

THEN 两个三角形全等

规则4： IF 一个三角形为等腰三角形

THEN 它的两底角相等

补充题 用5种搜索方法分别求解其搜索路径，并给出对应的OPEN表和CLOSED表



答：

1. **宽度优先搜索**

|  |  |  |
| --- | --- | --- |
| **搜索过程** | **OPEN表** | **CLOSED表** |
|  | B>A, H>A | A>NULL |
|  | H>A, C>B | A>NULL, B>A |
|  | C>B, G>H | A>NULL, B>A, H>A |
|  | G>H, D>C | A>NULL, B>A, H>A, C>B |
|  | D>C, F>G | A>NULL, B>A, H>A, C>B, G>H |
|  | F>G, E>D | A>NULL, B>A, H>A, C>B, G>H, D>C |

此时D节点的后继节点E为目标节点，搜索成功推出。

1. **深度优先搜索**

|  |  |  |
| --- | --- | --- |
| **搜索过程** | **OPEN表** | **CLOSED表** |
|  | B>A, H>A | A>NULL |
|  | C>B, H>A | A>NULL, B>A |
|  | D>C, G>C, H>A | A>NULL, B>A, C>B |
|  | E>D, F>D, G>C, H>A | A>NULL, B>A, C>B, D>C |

此时D节点的后继节点E为目标节点，搜索成功推出。

1. **等代价搜索**

|  |  |  |
| --- | --- | --- |
| **搜索过程** | **OPEN表** | **CLOSED表** |
|  | B>A, H>A | A>NULL |
|  | H>A, C>B | A>NULL, B>A |
|  | G>H, C>B | A>NULL, B>A, H>A |
|  | C>B, F>G, D>G | A>NULL, B>A, H>A, G>H |
|  | F>G, D>G | A>NULL, B>A, H>A, G>H, C>B |
|  | D>F | A>NULL, B>A, H>A, G>H, C>B, F>G |
|  | E>D | A>NULL, B>A, H>A, G>H, C>B, F>G, D>F |
|  | NULL | A>NULL, B>A, H>A, G>H, C>B, F>G, D>F, E>D |

此时E节点为目标节点，搜索成功推出。

1. **有序搜索**

|  |  |  |
| --- | --- | --- |
| **搜索过程** | **OPEN表** | **CLOSED表** |
|  | H>A, B>A | A>NULL |
|  | G>H, B>A | A>NULL, H>A |
|  | D>G, F>G, C>G, B>A | A>NULL, H>A, G>H |
|  | E>D, F>G, C>G, B>A | A>NULL, H>A, G>H, D>G |
|  | F>G, C>G, B>A | A>NULL, H>A, G>H, D>G, E>D |

此时E节点为目标节点，搜索成功推出。

1. **A\*算法**

|  |  |  |
| --- | --- | --- |
| **搜索过程** | **OPEN表** | **CLOSED表** |
|  | H>A, B>A | A>NULL |
|  | G>H, B>A | A>NULL, H>A |
|  | F>G, D>G, B>A, C>G | A>NULL, H>A, G>H |
|  | D>F, B>A, C>G | A>NULL, H>A, G>H, F>G |
|  | E>D, B>A, C>G | A>NULL, H>A, G>H, F>G, D>F |
|  | B>A, C>G | A>NULL, H>A, G>H, F>G, D>F, E>D |

此时E节点为目标节点，搜索成功推出。