



中山大學  
SUN YAT-SEN UNIVERSITY

## 实验报告

### 实验一：命题逻辑推理和最一般合一算法

姓 名	元朗曦
学 号	23336294
班 级	计算机八班
专 业	计算机科学与技术
学 院	计算机学院

## 一、算法原理

### 命题逻辑归结推理

1. 算法接收一个知识库（由多个子句组成，每个子句是包含字面量的元组）。将每个子句转换为可变列表，并记录初始子句及其编号。
2. 遍历所有子句对，对每一对检查其中是否存在互补字面量（例如“A”与“~A”）。若发现互补字面量，则从两个子句中分别移除对应的字面量，合并剩余字面量形成新的子句。
3. 对生成的新子句进行去重，并检查其是否已存在于知识库中；若没有，则添加到知识库，同时记录下分辨结果（包括使用的子句编号和省略的字面量标签）。
4. 当生成空子句时，表示发现矛盾，算法立即退出；否则继续对新的子句进行分辨，直到无法再生成新子句。

算法利用了归结原理，通过不断消去互补符号，如果最终能够推出空子句，则证明原知识库存在矛盾。

### 最一般合一算法

1. 通过函数将输入的字符串公式解析成谓词及其参数列表。使用辅助函数将带括号的参数字符串分解成单个参数，再递归解析嵌套的项或函数。
2. 约定无括号且长度大于1的标识符为变量。使用替换函数递归地将变量根据当前替换集进行替换，以保持统一进程中变量的一致性。在变量替换时，通过检查变量是否在目标项中出现，防止无限递归或循环引用。
3. 递归比较两个项：如果二者相同则直接返回；如果其中一个是变量且不在另一个项中，就将该变量替换为另一个项；如果两项均为复合项（函数形式），则先比较函数符号及参数个数是否一致，然后对各个对应参数进行统一；若统一过程中发现不匹配，则合一失败。
4. 将得到的替换映射转换为字符串形式输出，如果统一过程中出现异常，则返回空结果。

算法通过递归解析与统一、变量替换和循环检查，实现了两个一阶逻辑公式之间的最一般合一。

## 二、代码展示

见附件 week3-1.py 和 week3-2.py 文件。

## 三、实验结果

```
john@cobalt:~/Projects/sysu-assignments$ python -u
1 ('FirstGrade',)
2 (~FirstGrade, 'Child')
3 (~Child,)
4 R[1,2a] = ('Child',)
5 R[2b,3] = (~FirstGrade,)
6 R[1,5] = ()
```

图1 命题逻辑归结推理

```
john@coba1t:~/Projects/sysu-assignments$ python -u  
{'xx': 'b', 'yy': 'a'}  
{'zz': 'a', 'xx': 'f(a)', 'uu': 'g(yy)'}
```

图2 最一般合一算法