Web data mining部分：

第二章讨论部分：

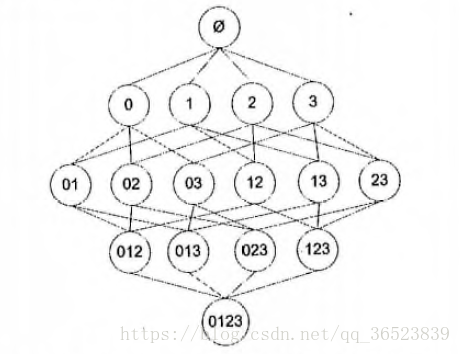
读书报告内容：

1. （必填）自己提出的问题的理解（罗列全部）：
2. 提出的问题1：

p35例21，在剪枝步，（1，2），（4），（6）被丢弃，因为1，4，6不在f3中，算法不是任何都嘛？

讨论后的理解：

这边应该是理解出了问题，算法描述的是任何一个不在频繁集，那么就可以丢弃，这和上周学习的AP算法也是相吻合的，如图：

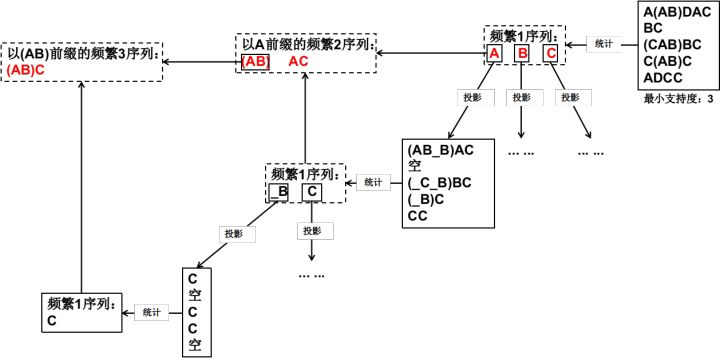


若12，不在频繁集中，那么012，123自然可以舍去，这和上周的ap算法是一脉相承的。

1. 提出的问题2：

prefixspan算法的核心思想是什么？

讨论后的理解：



其实本质上就是，先确定第一个项目作为前件，然后再确定后面的项目作为后件。假设前件为A，那么后件为B，再将A和B加起来，那么有两种加法，一个是{A，B}，还要一个是{A}{B}，这个过程可以递归进行。算法原理很简单，就是判定过程比较复杂。

1. （必填）别人提出的问题的理解（选择几个问题罗列，并给出理解）：

问题3：P34图2.13中合并步中，问什么说如果s1的第一个项目去掉后得到的余串和s1的最后一个项目去掉后的得到的余串相同就合并？

自己的理解：

因为对序列的挖掘，对序列进行合并，必然要考虑到序列的顺序，那么合并序列，自上而下的合并，一定是s1的除了开头和s2除了结尾的部分完全相同，才可以合并序列，这是为了保证序列的顺序。

问题4：P41最后一行，经过算法计算之后得到的序列模式集合为包括{<{30}，{40}>,<{40}，{30}>}这样的序列，算不算是矛盾呢？需不需要考虑删除呢？

自己的理解：

考虑一个现实的情况，我先买了一个玩具车，又买了电池，然后玩具车玩坏了，又买了个玩具车，那么就会出现上述的情况，实际上这种序列的发生恰恰说明了关系非常真实可靠，但是他们的序列就没有太大的意义了，那么可以变为关系模式挖掘继续分析。

问题6：P37页，为什么具有最小MIS的项目（设为x），不是s1中第一个项目，则s2中必然含有这个x? （例23？）

自己的理解：

因为书中明确说了前提条件是上面提到的问题，即为只有当序列s1的第一个项目或者序列s2的最后一个项目是各自序列中的唯一的MIS值最小的项目时才有可能发生。那么x不在s1的第一个项目中，那么x一定再s2之中。

问题6：P40写道PrefixSpan算法的主要优点使不生成候选序列，但在每一次对映射数据库进行匹配时，都需要遍历一次序列集，从时间复杂度问题上考虑，是不是并没有优于GSP呢？

自己的理解：

参考了刘立军老师的报告。GSP 算法类似于 Apriori 算法，是序列模式挖掘算法的基础，目前大多数的算法都是对此类算法的改进，GSP 算法有不可避免的劣势。PrefixSpan算法采用分治的思想，不断产生序列数据库的多个更小的投影数据库，然后在各个投影数据库上进行序列模式挖掘。但在实际应用中，在挖掘过程的不同阶段，数据集的特点，数据规模等因素可能不同，应根据各阶段的特点，选择与之相应的算法，使序列模式挖掘能达到更好的效果。

1. （必填）读书计划
2. 本周完成的内容章节：

（1）2.6到2.10完成并且梳理知识点，寻找问题，自己思考并且于小组会之前完成了自己的思考。

1. 西瓜书和李航的统计学习并未发现任何相关联章节。黑书的算法写的有点晦涩难懂，但是多读几遍，结合后面的例子仍然可以理解，同时参考了知乎上的文章，链接如下：

<https://www.zhihu.com/question/45592784>

关于本章两种算法复杂度的分析，书上唯有对比，但是翻阅了能找到 对比复杂度的报告和论文，论文标题为：



2、下周计划：5.1的阅读.

四、（选做）读书摘要及理解或伪代码的具体实现（读书摘要、伪代码的具体实现代码等可以写到这个部分）

1.代码实现

Prefixspan算法的代码实现找到频繁集：

