|  |
| --- |
| 西南民族大学学生实验报告  教学单位：计科学院 实验室名称：BS226 实验时间：2019年12月8日 |
| 实验项目名称：实验7—FP-growth 算法 实验成绩： 教师签名： |
| 实验项目报告内容（1、实验目的；2、实验主要过程与结果；3、分析讨论）。  **一、实验目的**   1. 学会构建FP树； 2. 学会从FP树中挖掘频繁项集。   **二、实验主要过程与结果**   1. 构建FP树    1. 创建FP树的数据结构   定义FP树中节点的类，该类的程序清单如图1所示。在控制台中测试该类，如图2所示。    图 1    图 2   * 1. 构建FP树   定义updateHeader()函数，用于更新头指针，建立相同元素之间的关系，函数代码和注释如图3所示。  定义updateTree()函数，用于更新FP-tree，函数代码和注释如图4所示。  定义createTree()函数，用于生成FP-tree，函数代码和注释如图5所示  为测试FP树，先用createInitSet()函数定义一个简单的数据集，并用updateHeader()函数实现数据集从列表类型到字典的类型转换，代码清单如图6所示。  在主函数中添加如图7所示测试语句，构建FP树，测试结果如图8所示。    图 3    图 4    图 5    图 6    图 7    图 8   1. 从一棵FP树中挖掘频繁项集    1. 抽取条件模式基   定义ascendTree()函数，用于迭代上溯整棵树，定义findPrefixPath()函数，用于遍历链表直到结尾，返回给定元素项结尾的所有路径，函数代码及注释如图9所示。  修改main函数中的测试代码，如图10所示，运行结果如图11所示。    图 9    图 10    图 11   * 1. 创建条件FP树   定义mineTree()函数，用于递归查找频繁项集，函数代码和注释如图12所示。  在主函数中加入如图13所示测试函数，运行结果如图14所示。    图 12    图 13    图 14   1. 示例：从新闻网站点击流中挖掘   在主函数中添加测试函数，如图15所示，运行结果如图16所示。    图 15    图 16  **三、分析讨论**   1. 2.2中的代码：   bigL = [v[0] for v in sorted(headerTable.items(), key=lambda p:p[1])]在python3中会报错：TypeError: unorderable types: treeNode() < treeNode()  原因：p[1]由两列元素构成，默认比较第一列元素，但是当第一列元素相等时会比较第二列元素，而第二列元素是treeNode类型，无法比较treeNode类型的数据。  解决办法：将p[1]转换为str类型或改成p[1][0]。 |

注：实验报告的内容及格式可由学院根据学科专业特点确定；全校各专业必须使用学校统一封面。