软件学院实验报告

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 学 号 | 542313460109 | | 姓 名 | 胡华吉 | 专业班级 | 软件工程23-01 |
| 课 程 | Python程序设计 | | 实验日期 | 2024.10.22 | 实验时间 | 第四大节 |
| 实验情况 | | | | | | |
| **实验四：正则表达式**  **一、实验目的**  （1）掌握正则表达式的构建；  （2）掌握正则表达式的匹配方法；  （3）通过选做任务，让学生掌握正则表达式在关键词统计和搜索中应用；  （4）创新思维：利用Python设计恰当的正则表达式思维训练。  **二、课程目标**  （1）掌握脚本语言程序设计的编程规范、基本知识、程序设计的基本理论、方法和应用；  （2）能够在软件开发过程中，针对特定需求，综合数据结构、算法复杂性分析、Python语言等知识设计实现相关算法，解决实际问题。能够独立设计相关算法，使用Python开发软件系统。  **三、实验任务**  1、请实现一个函数用来匹配包括'.'和'\*'的正则表达式。模式中的字符'.'表示任意一个字符，而'\*'表示它前面的字符可以出现任意次（包含0次）。在本题中，匹配是指字符串的所有字符匹配整个模式。例如，字符串"aaa"与模式"a.a"和"ab\*ac\*a"匹配，但是与"aa.a"和"ab\*a"均不匹配。  解题思路：判断字符串的第一位与正则表达式的前两位，然后进行递归。判断为True的结束条件是字符串s和正则表达式pattern的长度均为0；需要注意的是，s的长度不为0，pattern的长度为0返回False。下边的判断是针对pattern的第二位是否为“\*”开展的，如果为“\*”，则可能匹配0次、1次或多次；若不为“\*”，则只匹配s的第一位。  2、使用正则表达式匹配出百度主页代码中的地址，百度主页代码可以通过打开百度网页另存为文本文件得到。  3、a = "张明 98 分"，用re.sub，将98替换为100。  4、**高阶拓展**(选作)。利用正则表达式，统计歌曲《伟大的祖国》中“国”出现的频次，把含有“祖国”的那句话(前后用空格或逗号隔开)‬输出到显示器上。  **四、实验要求**  1. 对每个任务求解步骤是：算法思想、程序代码、运行结果的截图。  2. 提交实验报告。  **五、实验设计及结果**  **请在程序代码中通过注释加入自己的学号和姓名，将代码和运行结果截图放在此处。注意排版，保证图中字体能够看清。**   1. **：**   **算法思想：递归回溯，动态规划。**  **实现代码：**    **运行结果：**     1. **：**   **算法思想：暴力法**  **实现代码：**    **运行结果：**         1. **：**   **算法思想：暴力法**  **实现代码：**    **运行结果：**     1. **：**   **算法思想：暴力法**  **实现代码：**    **运行结果：**     1. **思考总结（本次实验的经验、教训，遇到的问题及解决方法，待解决的问题等）**  经验总结 理论与实践结合： 理论知识在实际应用中至关重要，通过编写实际代码，我更加理解了正则表达式的语法和使用场景。  递归思维的运用： 在实现正则表达式匹配功能时，递归思想的应用让我能够通过分解问题来逐步解决复杂的匹配逻辑。  正则表达式的灵活性： 正则表达式是处理文本数据强大的工具。在提取信息、替换文本等方面提供了高效的解决方案，使得代码简洁明了。  调试能力的提升： 在测试代码时，调试和跟踪每一步的逻辑变化帮助我发现并解决了不少错误，从而增强了我的调试能力。 教训反思 细节的重要性： 在处理正则表达式时，任何小的错误（如符号的遗漏或不当使用）都可能导致匹配失败。因此，认真审视每一部分是非常必要的。  性能考虑： 在设计递归算法时，需注意其性能问题，尤其是在输入数据量大的情况下。理解并应用动态规划可以有效优化递归算法的效率。  合理的注释和文档： 代码中添加注释和文档能帮助我在之后的调试和维护中更快地理解逻辑和思路，也能帮助其他人理解我的代码。 遇到的问题及解决方法 正则表达式不匹配： 在尝试提取URL时，发现某些边缘情况（如特殊字符）未被考虑。解决方法是重新审视正则表达式，调整模式以涵盖所有情况。  递归深度问题： 在实现字符串匹配时，输入较长的字符串导致递归深度超限。为了解决这个问题，我需要考虑使用动态规划或迭代的方法来避免递归调用。  调试难度： 初始实现的代码逻辑不够清晰，导致调试时遇到困难。通过逐步打印中间结果和条件判断，帮助我理清思路并定位问题。 待解决的问题 动态规划优化： 目前的递归解决方案在性能上存在提升空间，计划在未来实现一个基于动态规划的字符串匹配算法，以提升效率。  更复杂的正则表达式： 希望进一步学习和应用更复杂的正则表达式，以解决更具挑战性的文本处理问题。  处理多种语言文本： 研究如何处理包含多种语言或字符集的文本，以提高正则表达式在不同文本中的适用性和准确性。 | | | | | | |
| 实验报告成绩 | |  | | 指导老师 |  | |

注：1）专业班级按打印课表中名称填写；2）课程名称按课表中名称填写，不能简写；

3）实验日期格式示例：2024.03.09;4）实验时间格式示例：“第三大节”5）实验报告成绩按百分制评分。