

1 介绍

本报告旨在分析和比较硬币找零问题算法的理论和实际计算效率。我们使用了一个动态规划算法来求解限制硬币数量的硬币找零问题。

2 算法时间复杂度

动态规划算法的时间复杂度主要取决于硬币种类数 n 和目标金额 m 。该算法的时间复杂度为 $O(n \times m \times k)$ ，其中 k 为最大硬币数量。由于硬币种类和数量有限，我们可以期望算法在实际应用中表现出较好的效率。

3 实验设置

我们对不同的输入数据集进行了测试，以评估算法的实际执行时间。每个数据集包含了硬币的数量和目标金额。我们记录了处理每个数据集所需的时间，并与理论预期进行了比较。

4 结果与讨论

下表总结了算法对于不同数据集的执行时间和最小硬币数量：

目标金额	硬币数量	执行时间 (秒)	最小硬币数
11	3 2 1 0	7.13e-05	2
35	0 1 2 1	8.67e-05	2
40	5 0 3 1	0.0001048	7
30	2 2 0 1	6.28e-05	2
20	1 1 1 1	5.16e-05	-
52	4 3 2 1	0.0001086	6
157	23 4 6 2	0.0004591	-
104	7 5 3 2	0.0001069	13
313	3 2 10 8	0.0003321	23
112	1 5 2 3	8.09e-05	-

表 1: 算法执行时间和最小硬币数量

从结果中我们可以观察到，算法能够在多数情况下找到有效的最小硬币组合。对于无解的情况（标记为“-”），这通常意味着给定的硬币无法组合成目标金额。

5 结论

本报告展示了动态规划算法在处理硬币找零问题时的高效性和实用性。实验结果不仅与理论分析一致，而且还证明了算法能够有效地找到最小硬币数量的解决方案。