

实验报告

实验并测试Brute-Force、KMP、BM这三种字符串匹配算法.

实验环境

CPU: Intel Core i5-7200U

内存: 8GB

操作系统: Windows 10 64位教育版

编程语言: C++

运行方式: 运行exe目录下的exp.exe文件

算法分析

记模式串的长度为 m , 待匹配串的长度为 n .

Brute-Force算法

时间复杂度 $T = \Theta((n - m + 1)m)$.

KMP算法

算法分为两部分, Compute-Prefix-Function的时间复杂度 $T = \Theta(m)$, KMP-Matcher的时间复杂度 $T = \Theta(n)$, 总计时间复杂度 $T = \Theta(m + n)$.

BM算法

最大比较次数为 $3n - \Omega(n/m)$, 时间复杂度 $T = O(n)$.

结果分析

由于 n 较小时3种算法运行时间均小于 $1ms$, 时间误差较大, 所以取 $n = 10^7$, 在保持匹配串T中模式串P的出现频数 c 不变的基础上, 改变P的长度 m , 分别使用3种算法进行实验, 每组实验均进行5次取运行时间的平均值, 并记录其运行时间 (单位为ms) 如下表:

m	Brute-Force	KMP	BM
10^1	20	33	9
10^2	19	33	2
10^3	20	32	1
10^4	22	32	2

由数据表格可以初步看出，BM算法在运行速度上明显优于其他两种算法，且随着P的长度 m 增加有一定提升.在3种算法中KMP算法的效率最低，这可能与KMP算法对内存优化不友好的特点有关.

综上，在处理字符串匹配问题时应优先考虑使用BM算法.