

# KMP 算法

## 简介

KMP 算法是一种在字符串中进行高效模式匹配的算法，一种简单的算法是遍历字符串并进行模式匹配，如果匹配失败则从当前字符的下一个字符开始重新匹配。如下图中匹配字符 D 失败后，会从字母 B 开始重新检测是否匹配“ABCE”的模式。

A	B	C	D	E	F	G	H	I	J	K
A	B	C	E							

事实上这样的匹配方式会存在不必要的匹配过程。因为已经进行匹配的部分（ABC）中已经蕴含了下面两次匹配是 B 和 C，势必会匹配失败。因此 KMP 算法通过检测匹配模式的规律可以规避多余的匹配过程。

具体过程参考课件。

## 实验

本实验需要：

1. 实现上面提到的“简单”模式匹配算法
2. 实现 KMP 算法
3. 随机生成至少 4 组字符串（每组之间长度相差至少一个数量级，且每组字符串数量不少于 3 个）进行测试
4. 为每组字符串生成对应的匹配模式，其长度可以自行设计（如固定大小为 10），要求至少有 3 组不同的长度
5. 测试 KMP 算法是否相较简单版本的算法更高效，记录每种算法在不同字符串长度、匹配模式长度下的模式匹配的平均时长，将数据记录为表格并分析实验结果。（实验中对于测试数据的选取较为自由，进行实验并分析即可）

请将你的实验文档与代码和数据压缩后一同上传。