1. 串
2. 串的基本定义
3. 抽象数据类型

a∈character set，数据关系与线性表类似

1. 基本操作：

在线性表中往往以单个数据为操作单位，而在串中往往以数据的集合为操作单位

Attention.

1）字符串的比较（常规）：从首字母开始比较字典序

2）字符串取子串时需要考虑子串长度越界，长度为0等问题

3）字符串的替换时需要考虑是否可重叠，可重叠算法考虑一次替换后是否还存在需要替换的子串，不重叠算法仅进行一次替换

1. 串的表示与实现
2. 定长顺序存储表示
   1. 规定了串的最大长度
   2. 在串的连接中先判断连接时是否超限，然后再连接或截断
3. 堆分配存储表示
4. 块链存储表示
   1. 使用链表来存储串值，在链表的节点中存储串的一个子串，实际应用时也可自由设置节点大小。不同的子串节点间用指针连接，所以在编辑的过程中，只需要对块链内部进行操作，而不需要像顺序存储一样将此后每个位都移动
   2. 存储密度=数据元素所占存储位/实际分配的存储位
5. 串的模式匹配算法
6. 简单匹配算法 O(m\*n)

将子串不断与原串比较，一个个向后移动

1. 首尾匹配算法

将子串与原串从首尾开始比较，一个个向中间移动

1. **KMP匹配算法** O(m+n)

原理模式串中自我相似的子串在匹配时不必重复运算

对模式串通过一定的递推关系可以确定某个next数组值

Next数组其实是子串中最长相同前后缀的长度，但不能是字符串本身

递推算法确定next数组：若下一个字符与当前最长前后缀相同，则next数组+1

若下一个字符不同，向前查询第一个非0的next数，意为已有的相同前后缀长度，然后继续判断后一个字符是否相同，相同+1，不同则设为0（相当于再往前就找到0）

Next数组可确定指针的移动，在一次匹配不符合之后，主串中指针不变化，子串 中指针从头开始向后移动next数组值个字符然后重新开始匹配

最终在原串中其指针的移动不会后退，达成线性时间复杂度