

Homework 10

1. 试说明如何扩展 Rabin-Karp 算法用于处理以下问题: 在一个 $n \times n$ 的二维字符数组中搜索一个给定的 $m \times m$ 的模式。(该模式可以在水平方向和垂直方向移动, 但是不可以旋转。)
2. 对字母表 $\Sigma = \{a, b\}$, 画出与模式 ababbabbababbababbabb 对应的字符串匹配自动机的状态转换图
3. 描述解决这一问题的算法, 并对算法效果与复杂度进行分析 (不用完成 oj 上的题目!!!)
<https://202.38.86.171:1443/problem/E6-1>
4. 对于在无向图中寻找最长简单回路这一问题, 给出其形式化的定义并给出其相关的判定问题。另外, 给出与该判定问题对应的语言。

1. Rabin-Karp 算法本质上是一个 hash, 将 $P[1, \dots, m] \xrightarrow{f} p$

$$p = p[m] + d(p[m-1] + d(\dots)) \mod q$$

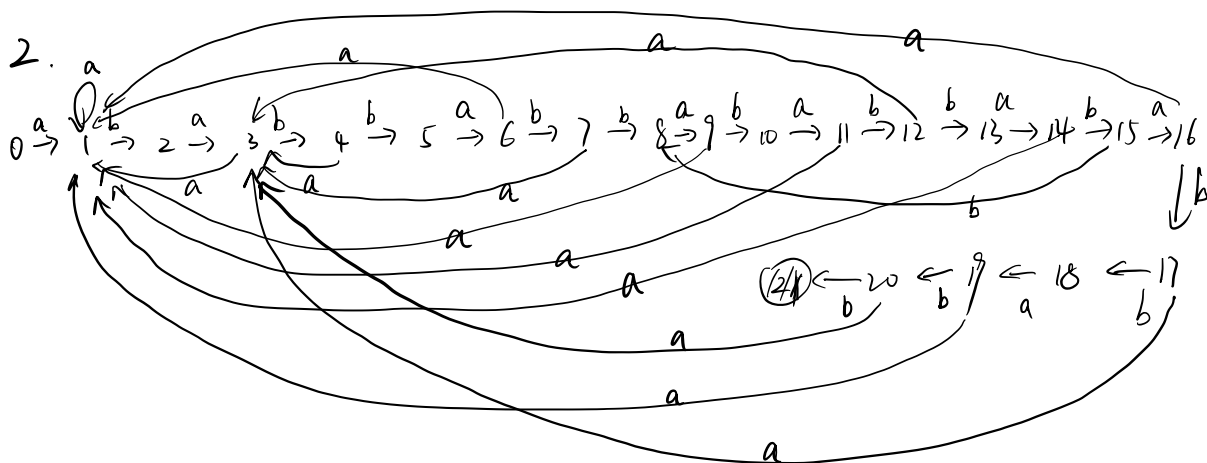
二维匹配可看成两次 hash, 具体步骤如下

① 预处理: $P[m][m] \xrightarrow{f} p'[m]$
 $T[m][n] \xrightarrow{f} T'_s[n] \quad (s=0)$

② 对 $p'[m]$ $T'_s[n]$ 进行 Rabin-Karp 算法

若找到匹配 退出, 否则转 ③, $s = n - m + 1$ 返回找到退出

③ 通过转移方程 求得 $T'_{s+1}[n]$; $s = s + 1$ 转 ②



这里与书中图32-1 一样省略了所有回到 0 的失败匹配

3. 对每组 (S, P) 进行如下算法。

$Alg(S, P)$:

$n = S.length$

$m = P.length$

$\pi = \text{compute-prefix-function}(P)$

$q = 0; cnt = 0;$

for $i = 1$ to n

while $q > 0$ and $P[q+1] \neq T[i]$

$q = \pi[q]$

if $P[q+1] == T[i]$

$q = q + 1$

if $q == m$:

$cnt++$

$q = \pi[q]$

return cnt

算法效果: 即将 KMP-match 算法中每次打印 offset 改为计数加 1, 最后返回计数即可。

复杂度: $\text{compute-prefix-function} : \Theta(m)$

Alg:

使用聚合分析: 只有执行 for 时 q 才会增加, 其它均减少

因为 $i = m$ 时就会终止 故 $q \leq i \leq n$ 又 $q \geq 0$

q 下降的次数最多为 n , 故 while 最多迭代 n 次, 运行时间 $\Theta(n)$

共运行次

总时间

$\Theta(T(m+n))$

4. 形式化定义: LONGEST-SIMPLE-CYCLE 是一个关系, 其将每个实例(即一个图)与这个图中最长的简单回路联系起来

判定问题: 给定 l , 判定实例 (图) 中是否有一个简单回路长度至少为 l
若是 返回 1 否则返回 0

语言: $\{ \langle G, k \rangle : G=(V, E) \text{ 是一个无向图, } k \geq 0 \text{ 是一个整数, } G \text{ 中存在一条长度至少为 } k \text{ 的简单回路} \}$