算法 week12

32.4-8

算法描述如下：

```
pi(1)=0

for k = 1 to m:

    if P[k+1]=P[pi(k)+1]:

        pi(k+1)=pi(k)+1

    else:

        pi(k+1)=0

    for a in ALPHABAT:

        if P[k+1] == a:

            delta(k, a)=delta(k+1, a)

        else:

            delta(k, a)=delta(pi(k), a)
```

显然，该算法经历两个循环，复杂度为$O(m|\Sigma|)$

32-1

a

1.首先计算整个P的前缀函数π。

2.对于某一个i，另$s = i - \pi(i)$，如果i mod s =0 且对于正整数r，有$p = i - r \cdot s > 0, p - \pi(p) = l$，则$\rho(P_i) = m/l$，否则$\rho(P_i) = 1$

运行时间为两部分，第一部分为计算π，第二部分为计算i的$\rho(P_i)$。时间复杂度为$O(m)$

b

$$
\begin{aligned}
P\left(\rho^*(P) \geq 2\right) &= \frac{1}{2} + \frac{1}{8} + \frac{1}{32} + \cdots \approx \frac{2}{3} \\
P\left(\rho^*(P) \geq 3\right) &= \frac{1}{4} + \frac{1}{32} + \frac{1}{256} + \cdots \approx \frac{2}{7} \\
P\left(\rho^*(P) \geq 4\right) &= \frac{1}{8} + \frac{1}{128} + \frac{1}{2048} + \cdots \approx \frac{2}{15} \\
P\left(\rho^*(P) = 1\right) &= \frac{1}{3} \\
P\left(\rho^*(P) = 2\right) &= \frac{8}{21} \\
P\left(\rho^*(P) = 3\right) &= \frac{16}{105} \\
\mathrm{E}\left[\rho^*(P)\right] &= 1 \cdot \frac{1}{3} + 2 \cdot \frac{8}{21} + 3 \cdot \frac{16}{105} + \ldots \approx 2.21 = O(1)
\end{aligned}
$$

c

观察匹配失败时，$s = s + \max(1, \lceil q/k \rceil)$，由于$q$是匹配成功的字符数量，而$k$是最大重复因子，所以右移后可以确保不错过匹配某一个字符串。可见，该算法正确。

算法中，$s = s + \max(1, \lceil q/k \rceil)$是唯一移动s的语句。其总共需要移动$O(\frac{\rho^*(P)n}{E(q)})$次，而每一次移动，都需要比较$E(q)$次，所以该过程的时间复杂度为$O(\rho^*(P)n)$。而计算$\rho^*(P)$的复杂度为$O(m)$，所以总的时间复杂度为$O\left(\rho^*(P)n + m\right)$