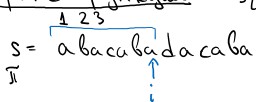


Префикс-функция $s[1..n]$



$P[i] = \max \{ k < i : s[1..k] = s[i-k+1..i] \}$
 Наибольший совпад. префикс строки $s[1..i]$,
 совп. с ее суффиксом

После построения $\#$ строки (алгоритм

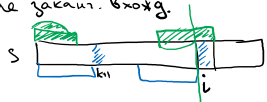
Кнута-Морриса-Прайта)
 KMP



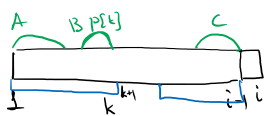
Посчит P -ф-ию за мин. время

```

P{1..n} = 0
P{2} = 0; k = 0
for i = 2..n:
    #k = P[i-1]
    while k > 0 and s[i] != s[k+1]:
        (*) -> k = P[k]
    if s[i] == s[k+1]:
        (***) -> k += 1
    P[i] = k
    
```



if $s[i] == s[k+1]: P[i] = k+1$
 Хотим перейти от k к наибольшему
 суфф. $s[1..i]$, совп. с ее префиксом,
 но поменьше



$A = C$ - хотим
 $B = C$, т.к. совп.
 $s[1..k]$
 и $s[i-k+1..i]$
 $\Rightarrow A = B$

Каждое вын. (*) уменьшает k на ≥ 1 .

Уменьшается оно 0 , и всегда ≥ 0 .

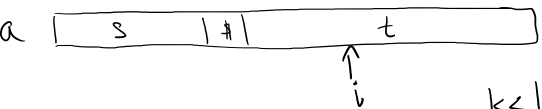
Увеличиваем k только в (**), то есть
 увелич. не больше чем на n за все время.

Следовательно, (*) вын. $\leq n$ раз.

Time = $O(n)$.

Mem = $O(n)$.

KMP: Time = $O(n+m) = O(|s|+|t|)$
 Mem = $O(n+m) \rightarrow O(n) = O(|s|)$



$k \leq |s|$, т.к. есть символ $\#$

$k = P[k]$

$s, t \quad |s| = n$

$P\{1..n\} = 0$

$P\{2\} = 0; k = 0$

$\#$ префикс-ф-ия где s

$k = 0$ $\#$ "обработан" $\#$

for c in t :

while $k > 0$ and $s[k+1] != s[i]$:

$k = P[k]$

← корректно!

if $s[i] == s[k+1]$:

$k += 1$

if $k == n$:

print ("boxgenie")