

Алгоритмы на строках

s - строка $s[0..n-1]$ $s[i] \in \Sigma$
 длина n английский

$s[l..r]$ - подстрока

$s[0..n]$ - префикс

$s[l..n]$ - суффикс

Поиск подстроки в строке (в тексте)

Ищем вхождение s в текст t.

- есть или нет?
- # вхождений
- позиции вхождений

$s = aba$ $|s| = n$
 $t = baabaabaaba$ $|t| = m$

Наивное решение

```
for i = 0..|t|-|s|:
    flag = True
    for j = 0..|s|-1:
        if s[j] != t[i+j]:
            flag = False
            break
    if flag:
        print("вхождение")
```

$O(nm)$
 $s = 'a' \cdot n; t = 'a' \cdot m$
 $(m-n) \cdot n \quad m = 2n \rightarrow n^2$
 Можно поспорить пример, когда вхождение не будет,
 а время $O(n^2)$.

Gr. find, strstr

Z-функция

$z[0..n]$
 $s = abacaba$
 $z = 00102001$

$z[i] = \begin{cases} 0, & i=0 \\ LCP(s, s[i..n]) \end{cases}$

$z[i] = \begin{cases} 0, & i=0 \\ \max\{k : s[0..k] = s[i..i+k]\} \end{cases}$

$s = aaaaaa$
 $z = 054321$

Поиск подстроки в строке: $a = s \# t$



Наивный алгоритм Z-функции

```
z = [0] * n
z[0] = 0
for i = 1..n-1:
    k = 0
    while s[i+k] == s[k]:
        k += 1
    z[i] = k
```

$O(n^2)$
 $s = aa...a$



Поиск Z-функции за мин. время

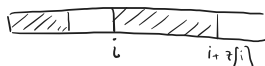
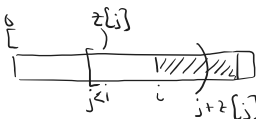
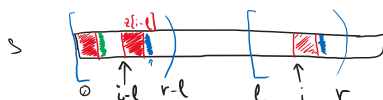
$z[i] = ?$ Знаем: $s[l..r] == s[0..r-l]$

1. $i-l+z[i-l] < r-l$ и $i+z[i-l] < r$
 2. $z[i] = z[i-l]$

2. иначе $(i+z[i-l] \geq r)$

3. $z[i] \geq r-i$
 Зависит от границ

```
z = [0] * n
z[0] = 0
l = r = 0
for i = 1..n-1:
    k = max(min(z[i-l], r-i), 0)
    while i+k < n and s[i+k] == s[k]:
        k += 1
    z[i] = k
    if i+k > r:
        l = i
        r = i+k
```



Time = $O(n + \# \text{встр. } s)$
 Для тех i, где кот. вст. сугубо s), (*) не вст. сугубо z: (*) вст.
 (xx) вст. $r = i+k$
 \forall вст. (*) $\rightarrow r+1$

Уточн.: $r = 0$
 В конце: $r \leq n$

$\text{Time} = O(n)$ Mem = $O(n)$

Поиск в строке: $O(n+m)$
 $O(|s|+|t|)$

Mem = $O(|s|+|t|)$