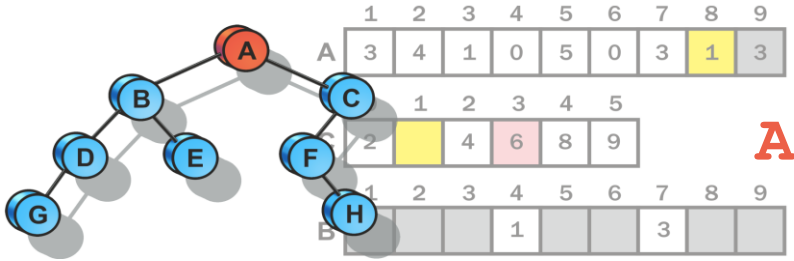
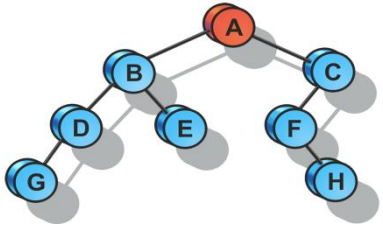
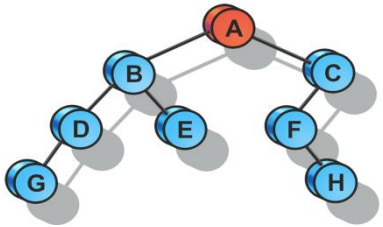


Поиск подстрок



Поиск подстрок 3.
Алгоритм Кнута — Морриса — Пратта



Вспомним простой алгоритм

. B A B C A A B A B C . .

A B C A A B D . . .

A B C A A B . . .

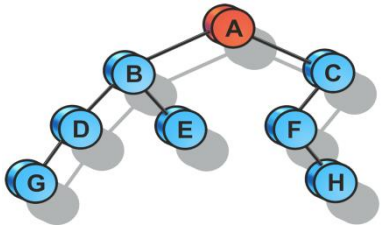
A B C A A . . .

A B C A . . .

A B C . . .

A B . . .

A . . .



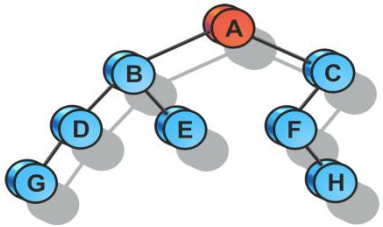
Большие сдвиги

. B A B C A A B A B C . .

A B C A A B D . . .

A B C . . .

A . . .



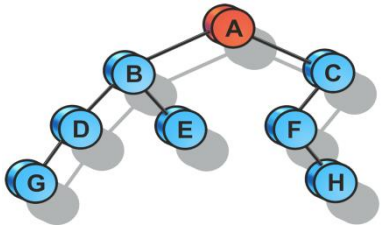
Префикс-функция

pref(S) = максимальный по длине собственный префикс строки S, являющийся также ее суффиксом

pref(“ABABAB”) = “ABAB”

pref(“ABCAAB”) = “AB”

pref(“AB”) = “”



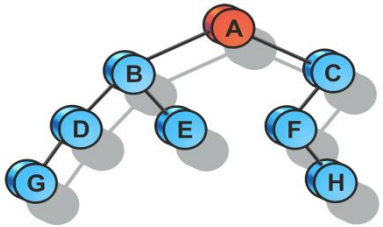
Большие сдвиги с использованием префикс-функции

. B A B C A A B A B C . .

A B C A A B D . . .

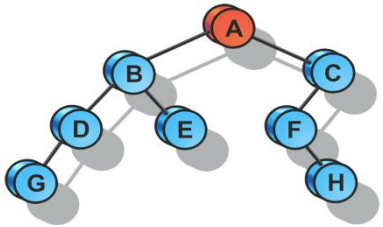
A B C . . .

A . . .



Префикс-функция для префиксов

- Посчитаем префикс-функцию для всех префиксов строки P.
- Префикс-функция от префикса – тоже префикс
- Достаточно хранить ее длину
- p[i] = |*pref*(P[0..i-1]))|

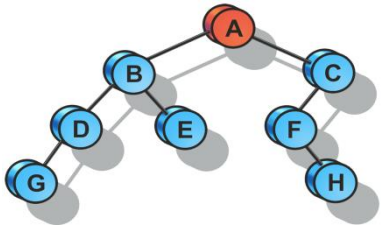


Пример

```
  ABACABABACB
p -00101232340
```

p[0]

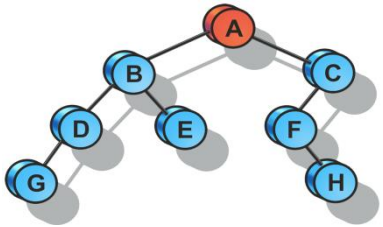
p[n]



Алгоритм поиска подстроки T

Состояние

```
                    i
S  .BABCAABABC..
T     ABCAAB...
                    j
```

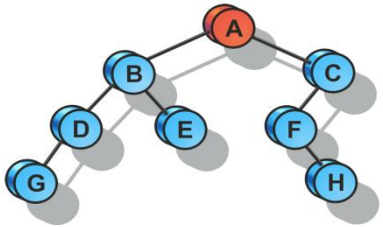


Алгоритм поиска подстроки T

Случай 1. S[i] == T[j]

```
                    i
S   .BABCAABABC..
T      ABCAABA...
                    j
```

```
i++
j++
```

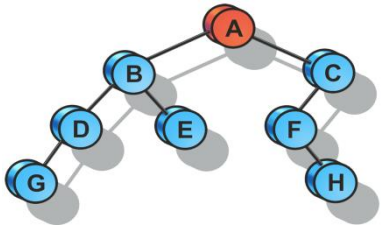


Алгоритм поиска подстроки T

Случай 2. S[i] != T[j], j>0

```
                i
S   .BABCAABABC..
T      ABCAABD...
          ABC...
           j
```

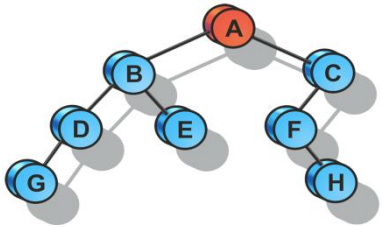
```
j = p[j]
```



Алгоритм поиска подстроки T

Случай 3. S[i] != T[j], j=0

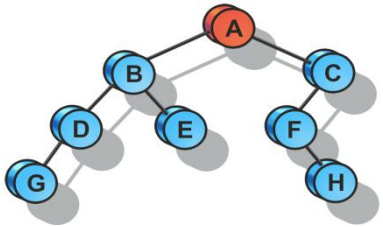
```
                  i
S   .BABCAABBAB..
T      ABCAABD...
           ABC...
             A...
               A..   i++
               j
```



Код

```
find(s, t: String):
  i = 0, j = 0
  while i < n && j < m:
    if s[i] == t[j]:
      i++, j++
    else:
      if j > 0:
        j = p[j]
      else:
        i++
  if j == m:
    return i - m
  else:
    return -1
```

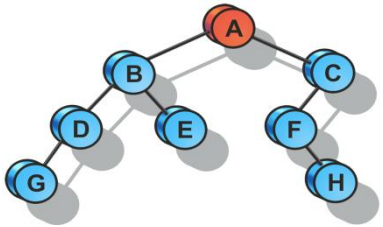
Время работы: O(N + M)



Алгоритм построения массива P

Состояние

```
                    i
T  .BABCAABABC..
T     ABCAAB...
                    j
```

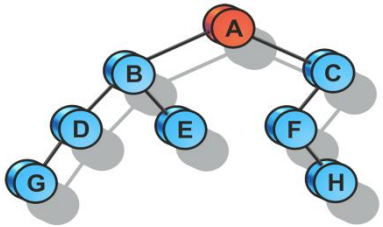


Алгоритм построения массива P

Случай 1. T[i] == T[j]

```
                  i
T  .BABCAABABC..
T     ABCAABA...
                  j
```

```
p[i+1] = j+1
i++
j++
```

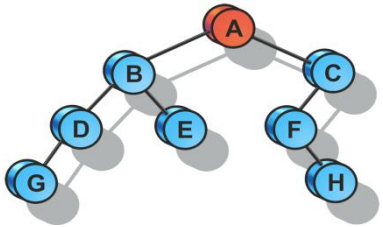


Алгоритм построения массива P

Случай 2. T[i] != T[j], j>0

```
                    i
T  .BABCAABABC..
T     ABCAABD...
            ABC...
            j
```

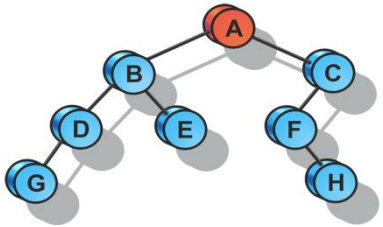
j = p[j]



Алгоритм построения массива P

Случай 3. T[i] != T[j], j=0

```
                  i
T   .BABCAABBAB..
T      ABCAABD...
          ABC...
             A...
             A..   p[i+1] = 0
             j     i++
```

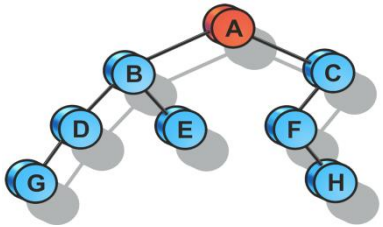


Алгоритм построения массива P

Пример

```
    i
ABACABABACB
  ABACABABACB
    j
p -0---------
```

```
p[i+1] = 0
i++
```

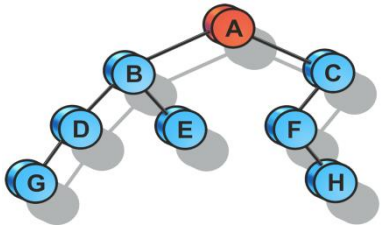


Алгоритм построения массива P

Пример

```
      i
ABACABABACB
   ABACABABACB
      j
p -00-------
```

```
p[i+1] = j+1
i++
j++
```

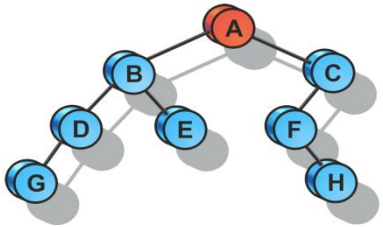


Алгоритм построения массива P

Пример

```
          i
ABACABABACB
   ABACABABACB
      j
p -0 0 1- - - - - - - -
```

```
j = p[j]
```

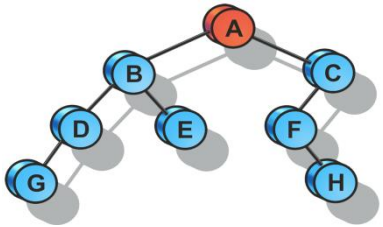


Алгоритм построения массива P

Пример

```
          i
ABACABABACB
    ABACABABACB
          j
p -001-------
```

```
p[i+1] = 0
i++
```

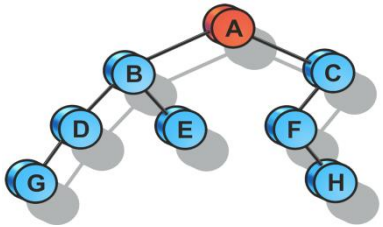


Алгоритм построения массива P

Пример

```
          i
ABACABABACB
    ABACABABACB
          j
p -0010-------
```

```
p[i+1] = j+1
i++
j++
```

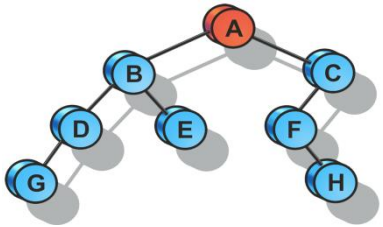


Алгоритм построения массива P

Пример

```
          i
ABACABABACB
    ABACABABACB
      j
p -00101------
```

```
p[i+1] = j+1
i++
j++
```

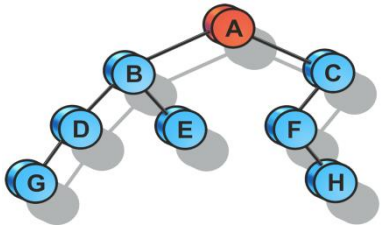


Алгоритм построения массива P

Пример

```
          i
ABACABABACB
   ABACABABACB
      j
p -0010120-----
```

```
p[i+1] = j+1
i++
j++
```

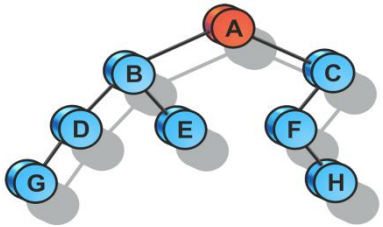


Алгоритм построения массива P

Пример

```
          i
ABACABABACB
    ABACABABACB
          j
p -0010123----
```

```
j = p[j]
```

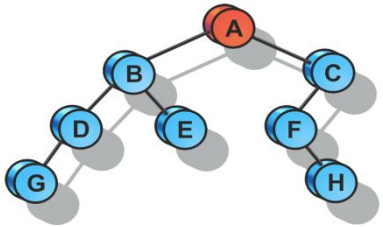


Алгоритм построения массива P

Пример

```
          i
ABACABABACB
      ABACABABACB
        j
p -0010123----
```

```
p[i+1] = j+1
i++
j++
```

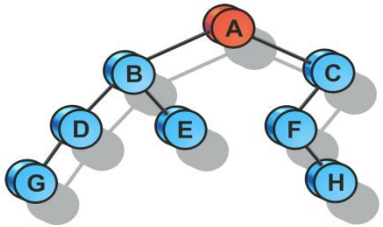


Алгоритм построения массива P

Пример

```
          i
ABACABABACB
      ABACABABACB
          j
p -0010123---
```

```
p[i+1] = j+1
i++
j++
```

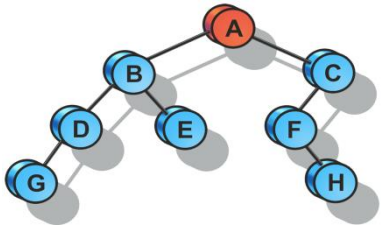


Алгоритм построения массива P

Пример

```
                    i
  ABACABABACB
           ABACABABACB
                    j
p -0010123230--
```

```
p[i+1] = j+1
i++
j++
```

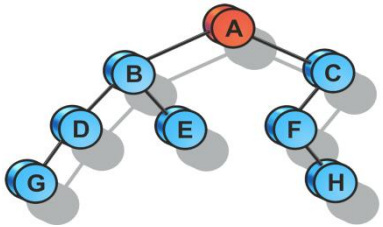


Алгоритм построения массива P

Пример

```
                        i
    ABACABABACB
            ABACABABACB
                j
p -0010123234-
```

```
j = p[j]
```

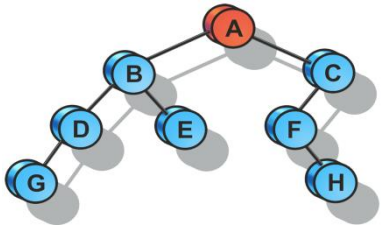


Алгоритм построения массива P

Пример

```
                    i
  ABACABABACB
                    ABACABABACB
                    j
p -0010123234-
```

```
p[i+1] = 0
i++
```

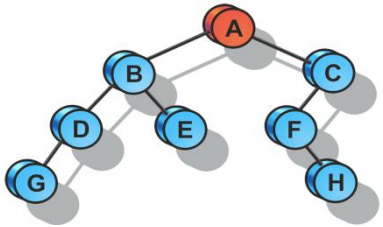


Алгоритм построения массива P

Пример

```
                  i
    ABACABABACB
                      ABACABABACB
                      j
p  -0010123234 0
```

```
p[i+1] = 0
i++
```



Код

```
buildP(t: String):
  i = 1, j = 0
  while i < n:
    if t[i] == t[j]:
      p[i + 1] = j + 1
      i++, j++
    else:
      if j > 0:
        j = p[j]
      else:
        p[i + 1] = 0
        i++
```

Время работы: O(N)