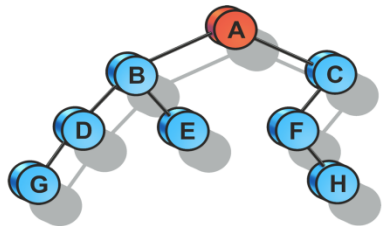


Алгоритмы программирования и структуры данных

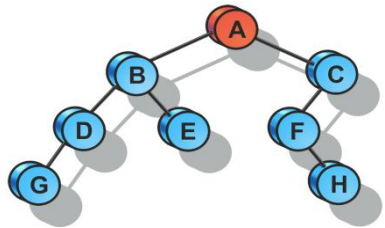
Введение в поиск подстрок



Алгоритмы программирования и структуры данных

Введение в поиск подстрок

Поиск подстрок (часть 1)

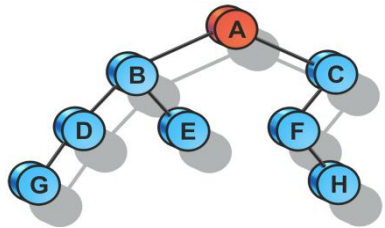


Строка

“ П Р О Г Р А М М И Р О В А Н И Е ”

1 2 3 4 5 6 ...

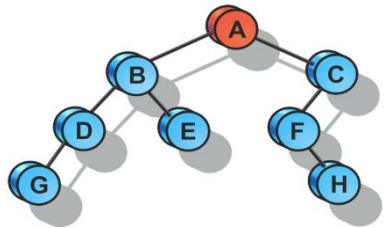
N-1



Конкатенация строк

“АЛГО” + “РИТМ” =

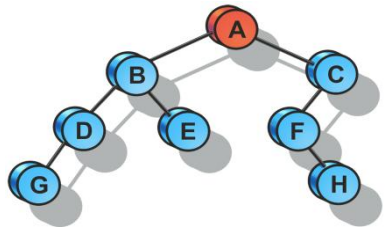
“АЛГОРИТМ”



Строка из нескольких слов

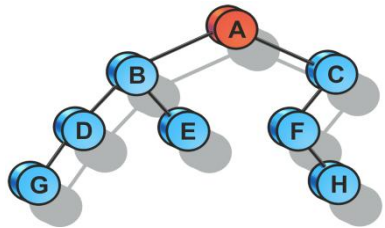
“АЛГОРИТМЫ И СТРУКТУРЫ ДАННЫХ”

“МОЙ ДЯДЯ САМЫХ ЧЕСТНЫХ ПРАВИЛ,
КОГДА НЕ В ШУТКУ ЗАНЕМОГ,
...”



Пустая строка

“ ”



Префиксы строки

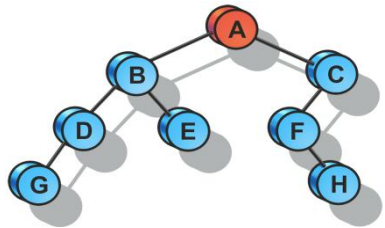
“ И Н Ф О Р М А Т И К А ”

“ И Н Ф О Р М А Т И К ”

“ И Н Ф О ”

“ И ”

“ ”



Суффиксы строки

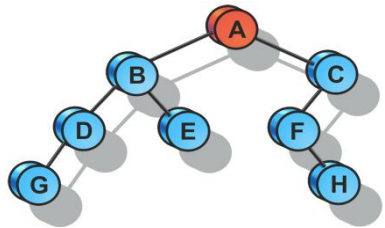
“ П О Д П Р О Г Р А М М А ”

“ П Р О Г Р А М М А ”

“ Г Р А М М А ”

“ М А ”

“ ”



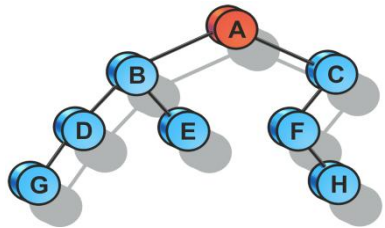
Подстроки

“ П Р О Г Р А М М И Р О В А Н И Е ”

“ Г Р А М М ”

“ Р О В ”

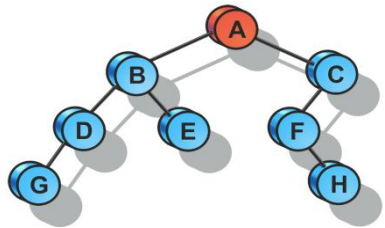
“ М И Р ”



Задача: найти T как подстроку S

S: “ П Р О Г Р А М М И Р О В А Н И Е ”

T: “ Р О В ”

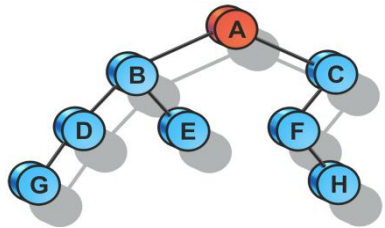


Простой алгоритм

ПРОГРАММИРОВАНИЕ
Р О В

ПРОГРАММИРОВАНИЕ
Р О В

ПРОГРАММИРОВАНИЕ
Р О В

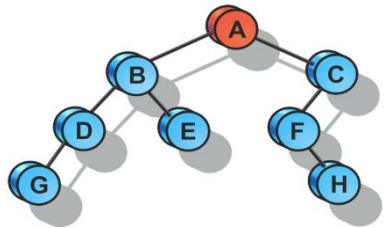


Простой алгоритм

ПРОГРАММИРОВАНИЕ
Р О В

ПРОГРАММИРОВАНИЕ
Р О В

ПРОГРАММИРОВАНИЕ
Р О В

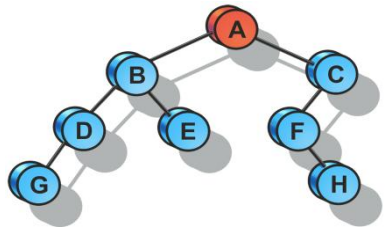


Простой алгоритм

ПРОГРАММИРОВАНИЕ
Р О В

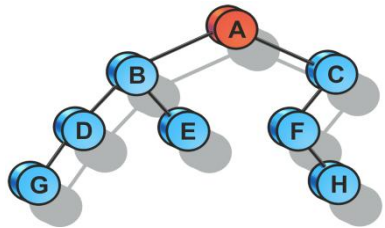
ПРОГРАММИРОВАНИЕ
Р О В

ПРОГРАММИРОВАНИЕ
Р О В



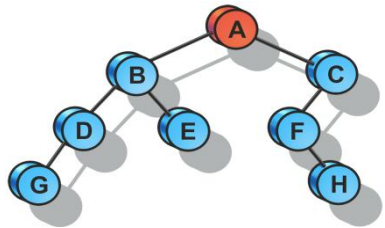
Простой алгоритм

ПРОГРАММИРОВАНИЕ
Р О В



Простой алгоритм. Код

```
find(s, t: String):  
    for i = 0 .. s.length - t.length:  
        ok = True  
        for j = 0 .. t.length - 1:  
            if s[i + j] != t[j]:  
                ok = False  
                break  
        if ok:  
            return i  
    return -1
```



Время работы алгоритма. Худший случай

S: “ A A A A A A A A A A A A A A A ”

T: “ A A A A A A B ”

Время работы: $O(NM)$