**Поиск подстроки в строке с помощью хеш-функции**

Поиск подстроки в строке - часто возникающая на практике задача. Поиск подстроки в строке обычной подстановкой к каждой позиции строки всей подстроки - метод неэффективный и вообще грустный. Я рассмотрю метод поиска с помощью хеш-функции - достаточно простой и быстрый.

Каждый символ имеет свой уникальный код от 0 до 255. Суть метода заключается в том, чтобы для подстроки подсчитать некоторую хэш-функцию (например сумму кодов всех символов в строке), затем посчитать ту же самую хэш-функцию для части строки, равной по длине подстроке, и, в случае совпадения хэш-функции, полностью сравнить его. Ускорение работы алгоритма связано с тем, что мы каждый раз не пересчитываем каждый раз хэш-функцию, а только отнимаем значение функции от самого "старого" символа и добавляем значение функции от следующего символа.

Пример:

Алфавит кодов:

Q = 1

W = 2

E = 3

R = 4

T = 5

Y = 6

Подстрока: QWERTY

Строка: QWERYTEWEQWERTY

Считаем хэш-функцию для подстроки: SS = 1+2+3+4+5+6+7 = 28

QWERYTEWEQWERTY

Считаем хэш-функцию для первых 6 символов строки: FS = 1+2+3+4+5+7+6 = 28

Проводим полное сравнение строк - строки не совпадают.

QWERYTEWEQWERTY

FS = 28 - [Q] + [E] = 28 - 1 + 3 = 30 - код не совпадает, сравнение не проводим.

QWERYTEWEQWERTY

FS = 30 - [W] + [W] = 30 - 2 + 2 = 30 - код не совпадает, сравнение не проводим.

QWERYTEWEQWERTY

FS = 30 - [E] + [E] = 30 - 3 + 3 = 30 - код не совпадает, сравнение не проводим.

QWERYTEWEQWERTY

FS = 30 - [R] + [Q] = 30 - 4 + 1 = 27 - код не совпадает, сравнение не проводим.

QWERYTEWEQWERTY

FS = 27 - [Y] + [W] = 27 - 6 + 2 = 23 - код не совпадает, сравнение не проводим.

QWERYTEWEQWERTY

FS = 23 - [T] + [E] = 23 - 5 + 3 = 21 - код не совпадает, сравнение не проводим.

QWERYTEWEQWERTY

FS = 21 - [E] + [R] = 21 - 3 + 4 = 22 - код не совпадает, сравнение не проводим.

QWERYTEWEQWERTY

FS = 22 - [W] + [T] = 22 - 2 + 5 = 25 - код не совпадает, сравнение не проводим.

QWERYTEWEQWERTY

FS = 25 - [E] + [Y] = 25 - 3 + 6 = 28 - код совпадает, полное сравнение совпадает. Ура!

Текст программы:

Program FSISHF; {поиск подстроки в строке}

Const

 NStr = 30000;

 NSub = 3000;

Var

 FStr : array[1..NStr] of char;

 FSub : array[1..NSub] of char; {substring}

 FSum, NSum : longint; {Контрольная сумма}

 Spec, Work : word;

 Flag : boolean;

Begin

 FSum := 0;

 NSum := 0;

 FillChar(FStr, SizeOf(FStr), 0);

 FillChar(FSub, SizeOf(FSub), 0);

 For Spec := 1 to NSub do

 FSum := FSum + Ord(FSub[Spec]);

 For Spec := 1 to NSub do

 NSum := NSum + Ord(FStr[Spec]);

 For Spec := NSub to NStr do begin

 If NSum = FSum then begin

 Flag := true;

 For Work := 1 to NSub do

 If FSub[Work] <> FStr[Spec - NSub + Work] then begin

 Flag := false;

 break;

 end;

 If Flag = true then begin

 Writeln ('substring starts at position: ', Spec - NSub);

 Halt;

 end;

 end;

 NSum := NSum + Ord(FStr[Spec + 1]) - Ord(FStr[Spec - NSub + 1]);

 end;

 Writeln('no such substring');

end.