

```
1 Program rabin;  
2 {
```

```
    MSP, 09.02.06, 12:44
```

Алгоритм Рабина-Карпа

Дано:

T - текст из n символов.

T [i] - его i-тый символ.

S - слово-строка-образец, которое ищем в T, его длина m

S [i] - i-тый символ слова-строки-образца.

Задача.

Узнать, входит ли S в T, и если входит, то начиная с какого символа текста T.

Поставим в соответствие каждой строке некоторое уникальное число, и вместо того, чтобы сравнивать сами строки, будем сравнивать числа, что намного быстрее.

Проблема в том, что искомая строка может быть длинной, строк в тексте тоже немало. А так как каждой строке нужно сопоставить УНИКАЛЬНОЕ число, то и чисел должно быть много, а стало быть, числа будут большими (порядка  $D * m$ , где D - количество различных символов, m - длина слова-строки), и работать с ними будет также неудобно.

Выход из положения предлагает алгоритм Рабина-Карпа.

Прочтите внимательно нижележащий алгоритм.

Как вы увидели, нам все-таки приходится производить сравнение строк по-символьно, но так как "холостых" срабатываний будет немного (в  $1/P$  случаях), то ожидаемое время работы малое. Строго говоря, время работы есть  $O(m + n + mn/P)$ , где  $mn/P$  досаточно невелико, так что сложность работы почти линейная.

Этот алгоритм подходит для работы с очень длинными строками.

```
}
```

```
31 Uses Crt;
```

```
32 Const
```

```
33 {  
    простое число выбираем специально большим, чем больше это  
    число, тем быстрее будет работать программа  
}
```

```
37 P : longint = 7919; { 1000-е простое число }
```

```
38 D : longint = 256; { число разных символов }
```

```

39  Var
40      T { данный текст }
41      : array [1..40000] of char;

42      S { искомая слово-строка-образец }
43      : array [1..10000] of char;

44      i, j, { индексы T и S }
45      n, { длина текста }
46      m, { длина образца }
47      w, { уникальное число для текста }
48      v, { уникальное число для образца }
49      k
50      : Longint;

51  Begin
52      ClrScr;

53      {
54          ввод текста T и образца S
55      }
56      Write ('Введите длину текста T: ');
57      ReadLn (n);

58      if n > 10
59          then
60              Exit;

61      WriteLn ('Введите текст T:');
62      for i := 1 to n do
63          Read (T [i]);

64      ReadLn;

65      Write ('T: ');
66      for i := 1 to n do
67          Write (T [i]);
68      WriteLn;

69      WriteLn;
70      Write ('Введите длину образца S: ');
71      ReadLn (m);

72      if m > 8
73          then
74              Exit;

75      WriteLn ('Введите текст S:');
76      for i := 1 to m do
77          Read (S [i]);

```

```

78     ReadLn;

79     Write ('S: ');
80     for i := 1 to m do
81         Write (S [i]);
82     WriteLn;

83     {
        посчитаем уникальные числа для текста и образца,
        причем для текста, только для первых m символов
    }

87     w := 0;
88     v := 0;
89     for i := 1 to m do
90         begin
91             w := (w * D + Ord (T [i])) mod P; { Ord () - код символа }
92             v := (v * D + Ord (S [i])) mod P;
93         end;

94     WriteLn ('уникальное число для T - w: ', w);
95     WriteLn ('уникальное число для S - v: ', v);

96     {
        найдем  $k = D * m - 1 \bmod P$ 
    }

99     k := 1;
100    for i := 1 to m - 1 do
101        k := k * D mod P;

102    WriteLn ('k = ', k);

103    {
        посмотрим текст на совпадение с образцом с символа,
        стоящего на m + 1 месте в T
    }

107    for i := m + 1 to n + 1 do
108        begin
109            {
                если уникальные числа совпадают, то строки принадлежат
                одному типу и надо проверить, совпадают ли они
            }

113            if w = v
114                then
115                    begin
116                        { проверим на совпадение первые m - 1 символ }
117                        j := 0;
118                        while (j < m)
119                            and
120                                (S [j + 1] = T [i - m + j])
121                            do

```

```

122             Inc (j);

123             { а последние символы совпадают? }
124             if j = m
125             then
126                 WriteLn ('Образец входит в текст начиная с ',
127                     i - m, '-ого символа' );
128             end;

129         {
            если уникальные числа не совпадают,
            то сдвинемся на 1 позицию в T и вычислим для нового
            отрезка, длиной в m, соответствующее уникальное число
        }
134         if i <= n
135         then
136             begin
137                 w := (D * (w + P - (Ord (T [i - m]) * k mod P)) +
138                     Ord (T [i])) mod P;

139                 WriteLn ('new_w = ', w);
140             end;
141         end;

142     ReadLn;
143     End.

```

Listing данной задачи опубликован в сети Internet по адресу  
<http://www.Best-Listing.ru/color-1-task-378.html>

Sergey Mitrofanov, 25.11.13, 22:44

E-mail: infostar@mail.ru

© <http://www.Best-Listing.ru/>, 2006–2013