

Ульянцев Владимир Игоревич,
Царев Федор Николаевич

ЗАДАЧА «АНТИПАЛИНДРОМ»

Этой статьей мы продолжаем цикл публикаций олимпиадных задач для школьников по информатике и программированию с разборами. Решение таких задач и изучение разборов поможет Вам лучше подготовиться к олимпиадам по информатике.

В этой статье рассматривается задача «Антипалиндром», которая предлагалась в первой Интернет-олимпиаде сезона 2008–2009. Интернет-олимпиады по информатике проводятся Санкт-Петербургским государственным университетом информационных технологий, механики и оптики в двух номинациях – базовой и усложненной. Базовая номинация рассчитана на начинающих участников олимпиад, поэтому в ней предлагаются более простые задачи, а в усложненной номинации предлагаются задачи уровня городских и всероссийских командных олимпиад по программированию. Сайт этих олимпиад находится по адресу <http://neerc.ifmo.ru/school/io/>.

УСЛОВИЕ ЗАДАЧИ

Палиндромом называют строку, читающуюся одинаково с обеих сторон.

Задана строка s . Найдите ее наибольшую по длине подстроку, не являющуюся палиндромом.

Формат входного файла

Входной файл содержит строку s . Она состоит только из строчных букв латинского алфавита, не пуста, а ее длина не превышает 100000 символов.

Формат выходного файла

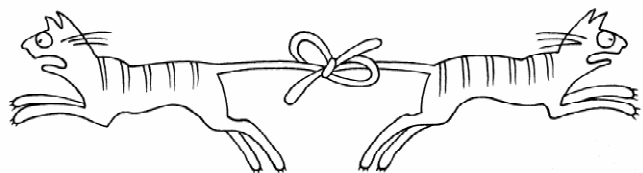
В выходной файл выведите ответ на задачу. Если все подстроки s являются палиндромами, выведите в выходной файл NO SOLUTION.

Пример входных и выходных данных

<code>antipali.in</code>	<code>antipali.out</code>
abba	abb

РАЗБОР ЗАДАЧИ

Заметим, что число подстрок строки длины n равно $1 + 2 + \dots + n = n(n + 1)/2$. Это объясняется тем, что строка длины n содержит одну подстроку длины n , две подстроки длины $(n - 1)$, ..., n подстрок длины 1. Таким образом, время работы решения, основанного на переборе всех подстрок, будет пропор-



ционально как минимум n^2 . Такое решение не укладывается в ограничения по времени. Пример такого решения на языке Pascal приведен в листинге 1.

Для того чтобы получить более быстрое решение, необходимо детальнее изучить свойства подстрок, не являющихся палиндромами. Во-первых, сама строка s может не быть палиндромом. Тогда она и является ответом для задачи. Во-вторых, может оказаться так, что все подстроки строки s являются палиндромами. Такое может быть только в том случае, если все символы s равны между собой.

Докажем теперь, что если ни один из этих двух случаев не реализуется, то строка, получаемая из s отбрасыванием первого символа не является палиндромом. И так, строка s является палиндромом, но не все ее символы равны между собой. Обозначим как t строку, получающуюся из s отбрасыванием первого символа. Докажем с помощью метода «от противного», что t не является палиндромом.

Для этого предположим, что $t = s_2 s_3 \dots s_n$ является палиндромом. Отсюда следует, что

ее первый и последний символ равны между собой, то есть $s_2 = s_n$. Из того, что s является палиндромом, следует, что $s_1 = s_n$. Таким образом, получаем, что $s_1 = s_2 = s_n$. Далее, из того, что s является палиндромом, также следует, что $s_{n-1} = s_2$. Значит, $s_1 = s_2 = s_n = s_{n-1}$. Из того, что t является палиндромом, следует, что $s_3 = s_{n-1}$. Значит, цепочка равенств расширяется далее: $s_1 = s_2 = s_n = s_{n-1} = s_3$. Продолжая это рассуждение, получаем, что все символы строки s равны между собой, что противоречит предположению. Значит, доказано, что строка t не является палиндромом.

Это рассуждение приводит нас к следующему решению. Возможны три случая:

- все символы строки s равны между собой – тогда ответ NO SOLUTION;
- строка s не является палиндромом – тогда ответом является сама строка s ;
- иначе ответом является строка t , получаемая из s отбрасыванием первого символа.

В листинге 2 приведена программа на языке Pascal, реализующая это решение.

Листинг 1. Первый способ решения

```

var
  n, j, pos, len : integer;
  s : string;
begin
  reset(input, 'antipali.in');
  rewrite(output, 'antipali.out');

  read(s);
  n := length(s);

  for len := n - 1 downto 0 do
    for pos := 1 to n - len do
      for j := 0 to len div 2 do
        if (s[pos + j] <> s[pos + len - j]) then begin
          writeln(copy(s, pos, len + 1));
          halt;
        end;
      end;
    end;
  end;

  writeln('NO SOLUTION');
end.

```

Листинг 2. Второй способ решения

```
var
  s : string;
  n, i : longint;
  allSame, pali : boolean;

begin
  reset(input, 'antipali.in');
  rewrite(output, 'antipali.out');

  read(s);
  n := length(s);

  allSame := true;
  pali := false;

  for i := 2 to n do begin
    if (s[i] <> s[i - 1]) then begin
      allSame := false;
    end;
  end;

  for i := 1 to n do begin
    if (s[i] <> s[n - i + 1]) then begin
      pali := true;
    end;
  end;

  if (allSame) then begin
    writeln('NO SOLUTION');
  end else if (not pali) then begin
    for i := 1 to n - 1 do begin
      write(s[i]);
    end;
  end else begin
    writeln(s);
  end;
end.
end.
```

*Члены жюри Интернет-олимпиад
по информатике базового уровня:*

*Ульянцев Владимир Игоревич,
студент пятого курса кафедры КТ
СПбГУ ИТМО, член жюри ВКОШП,*

*Царёв Федор Николаевич,
аспирант кафедры КТ СПбГУ
ИТМО, чемпион мира по
программированию среди студентов
2008 года.*

© Наши авторы, 2011.
Our authors, 2011.