



Владислав Сергеевич Попов

Учитель информатики ГБОУ Цифровая школа,
старший преподаватель кафедры «Информационные системы
и телекоммуникации» МГТУ им. Н. Э. Баумана

Переводим псевдокод, или Решение задачи № 12 ЕГЭ по информатике на обработку строк

Решим задание № 12 демонстрационного варианта ЕГЭ по информатике с помощью программ на языках программирования Python, C++, Pascal. Сокращённая формулировка задания:

Какая строка получится в результате применения приведённой ниже программы к строке, состоящей из 70 идущих подряд цифр 8? В ответе запишите полученную строку.

НАЧАЛО

ПОКА нашлось (2222) ИЛИ нашлось (8888)

 ЕСЛИ нашлось (2222)

 ТО заменить (2222, 88)

 ИНАЧЕ заменить (8888, 22)

КОНЕЦ ЕСЛИ

КОНЕЦ ПОКА

КОНЕЦ

Ответ: 22.

Решение задания на языке программирования Python

```
s = '8' * 70
while '2222' in s or '8888' in s:
    if '2222' in s:
        s = s.replace('2222', '88', 1)
    else:
        s = s.replace('8888', '22', 1)
print(s)
```

В первой строке программы при помощи умножения строки '8' на число 70 создана строка из 70 цифр «8».

Для проверки наличия подстроки в строке s применяется оператор in, но также можно применить метод find, который возвращает индекс начала подстроки или -1, если подстрока не найдена. Таким образом более простую запись условия '2222' in s можно заменить на s.find('2222') != -1:

```
s = '8' * 70
while s.find('2222') != -1 or s.find('8888') != -1:
    if s.find('2222') != -1:
        s = s.replace('2222', '88', 1)
    else:
        s = s.replace('8888', '22', 1)
print(s)
```

Замена строки s выполняется с помощью присваивания в строку s результата работы метода replace. Запись s = s.replace('2222', '88', 1) заменяет в строке s первое слева вхождение подстроки «2222» на подстроку «88» один раз. При работе с методом replace необходимо использовать оператор присваивания для изменения строки и указать третий аргумент метода – количество производимых замен в строке – равным 1.

В последней строке программы по завершении цикла while, работающего, пока его условие истинно, выполняется печать изменённой строки s.

Решение задания на языке программирования C++

```
#include <iostream>
#include <string>
using namespace std;
int main()
{
```

```

string s(70, '8');
while (s.find("2222") != string::npos ||
       s.find("8888") != string::npos){
    if (s.find("2222") != string::npos){
        s.replace(s.find("2222"), 4, "88");
    }
    else{
        s.replace(s.find("8888"), 4, "22");
    }
}
cout << s << endl;
}

```

В первой строке функции main создаётся строка типа `string` с помощью конструктора, принимающего два параметра: первый – количество повторений символа, а второй – сам символ. Таким образом, при помощи вызова конструктора `string s(70, '8')` будет создана строка `s`, состоящая из 70 символов «8». Напомним, что символы в языке C++ указываются в апострофах (одинарных кавычках), а строки – в кавычках.

Для нахождения подстроки в строке применяется метод `find`, который возвращает индекс первого слева вхождения подстроки или константу `string::npos` (обычно равную -1), если подстрока не была найдена. Для замен в строке применяется метод `replace`, принимающий в качестве трёх аргументов индекс подстроки для замены, длину подстроки для замены и новую подстроку для вставки: например, вызов метода `s.replace(s.find("8888"), 4, "22")` заменит первое слева вхождение подстроки «8888» из 4 символов на подстроку «22».

Решение задания на языке программирования Pascal

Решение с использованием подпрограмм для работы со строками в среде PascalABC.NET:

```

var
    s: string;
    p: integer;
begin
    s := '8' * 70;
    while (pos('2222', s) <> 0) or (pos('8888', s) <> 0) do
    begin
        if pos('2222', s) <> 0 then
        begin
            p := pos('2222', s);
            delete(s, p, 4);
            insert('88', s, p);
        end
    end
end

```

```

        end
    else
    begin
        p := pos('8888', s);
        delete(s, p, 4);
        insert('22', s, p);
    end;
end;
writeln(s);
end.

```

В данном решении были использованы следующие подпрограммы: pos (подстрока, строка) для поиска позиции первого вхождения подстроки в строке, если подстрока не найдена, функция pos возвращает значение 0, значение позиции первого символа в строке принято за 1; delete (строка, позиция, количество) для удаления заданного количества символов из строки, начиная с заданной позиции; insert (подстрока, строка, позиция) для вставки заданной подстроки в строку, начиная с заданной позиции. Совмещение функций delete и insert потребовалось ввиду отсутствия функции замены (replace) для строк в языке программирования Pascal, данная функция присутствует лишь в некоторых его диалектах.

Следует отметить, что оператор умножения для строк и чисел перегружен в PascalABC.NET, но не является перегруженным и не может быть применён во множестве других компиляторов. В случае использования другого компилятора для инициализации элементов строки можно воспользоваться циклом for. Будьте осторожны: размер строк типа string в некоторых компиляторах существенно ограничен и может быть меньше, чем требуемый в задании.

Также возможно решить данную задачу с использованием методов класса String в среде PascalABC.NET, при этом решение получится более простым:

```

var
    s: string;
begin
    s := '8' * 70;
    while s.Contains('2222') or s.Contains('8888') do
    begin
        if s.Contains('2222') then
            s := s.Replace('2222', '88', 1)
        else
            s := s.Replace('8888', '22', 1);
        end;
    writeln(s);
end.

```

Метод `Contains` возвращает значение `True`, если искомая подстрока входит в строку, метод `Replace` возвращает строку, полученную из исходной строки заменой указанного в третьем аргументе количества вхождений первого аргумента на второй аргумент. Метод `Contains` может быть заменён на метод нахождения индекса первого вхождения подстроки в строку: `s.Contains('2222')` можно заменить на `(s.IndexOf('2222') <> -1)`. Как и при решении задания на языке программирования Python, при вызове метода `Replace` необходимо не забыть указать количество замен равным 1 и присвоить возвращённое методом значение в строку – иначе строка не изменится.

Универсальный совет, подходящий для любого из рассмотренных решений: если выводить строку `s` после каждого изменения в теле цикла, то можно увидеть, каким образом будет работать алгоритм из задания, а также найти и исправить ошибки в написанном коде. Для этого оператор/функцию вывода нужно поместить в качестве последней инструкции в теле цикла.

В статье был рассмотрен простой способ решения задачи № 12 ЕГЭ по информатике на языках программирования Python, C++, Pascal, однако даже опытные программисты могут ошибаться, поэтому для контроля правильности ответа в первую очередь следует решить задачу самостоятельно, и только потом проверить полученный ответ на компьютере.

МУДРЫЕ МЫСЛИ

Нельзя быть настоящим математиком, не будучи немного поэтом.

...Между математикой и естественными науками должны быть установлены более глубокие взаимоотношения, чем те, которые имели бы место, если бы, например, физика видела в математике лишь вспомогательную дисциплину, пусть даже необходимую, а математика рассматривала вопросы, выдвигаемые физиками, только как обильное собрание примеров для своих методов.

На вопрос: «Можно ли действительно получить что-либо непосредственно применимое из тех абстрактных теорий, которыми предпочтительно занимаются теперешние математики?» – я могу ответить, что греческие математики изучили свойства конических сечений чисто умозрительным путём задолго до того, как кто-либо мог предугадать, что эти кривые представляют собой пути, по которым движутся планеты, и я верю, что будет найдено ещё много функций с такими свойствами, как, например, знаменитые тета-функции Якоби, с помощью которых можно, с одной стороны, узнать, на сколько квадратов разлагается любое заданное число, которые позволяют спрямить дугу эллипса, и, с другой стороны, дают возможность найти истинный закон колебаний маятника.

Карл Вейерштрасс, немецкий математик