

Информатика

Ларина Элла Семёновна

*Учитель информатики высшей категории
МОУ «Лицей №2» г. Волгограда.*



Как мы делили апельсин

Вооружившись лишь небольшим запасом знаний по программированию (включающем в себя знания по реализации арифметических операций и базовых алгоритмических конструкций) можно как успешно сдать ЕГЭ, так и попробовать свои силы в олимпиадном движении!

«Мы делили апельсин, много нас, а он один!». Кто участвовал в дележе апельсина в известной всем песенке уже и не вспомнить. Да и интереса особого они у нас не вызывают – в отличие от апельсина, дольки которого пересчитать не помешало бы!

Итак, есть апельсин, состоящий из N долек, которые необходимо разделить на K равных частей (ломать дольки нельзя!).

В решении данной задачи на языке программирования Паскаль необходимо использовать арифметические операции целочисленного деления **div** и **mod**:

- при помощи операции **div** мы получим количество целых долек, приходящееся

на одного из участников деления,

- при помощи **mod** узнаем, сколько долек ушло в остаток (нацело не поделилось).

А если предположить, что каждый, получивший свою долю, придя домой, поделил её между K домочадцами, каждый из домочадцев поделил полученное между K своими товарищами, которые, в свою очередь, также разделили в своей семье горку апельсиновых долек... Циклический процесс деления продолжится до тех пор, пока на руках многочисленных участников не окажется такое количество долек, которое невозможно поделить между K любителями сладкого. Найдём это количество (при помощи цикла **While**):

двух ячеек памяти нужно с использованием третьей ячейки **z** (так же, как

и для переливания содержимого двух стаканов – через третий).

```
var x,y,z,k:integer;
begin
  readln (x,y);
  if x<y then begin
    z:=x; x:=y; y:=z;
  end;
  while ((x>0)and(y>0)) do
  begin
    k:=k+ (x div y);
    x:=x mod y;
    z:=x; x:=y; y:=z;
  end;
  writeln (k);
end.
```

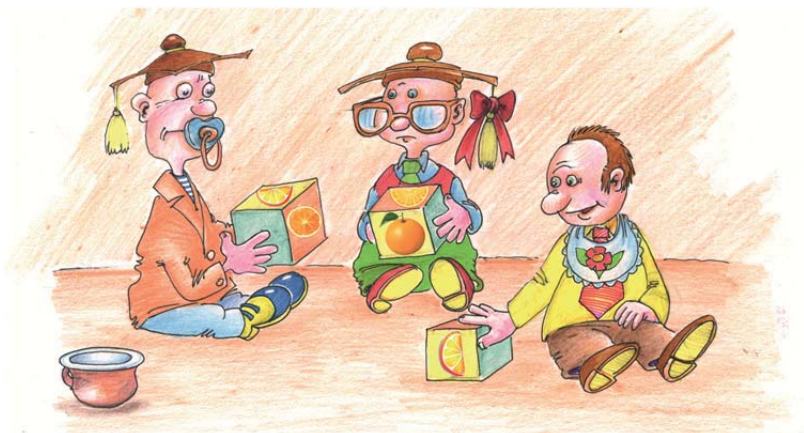
Серым цветом так же, как и в предыдущей задаче, выделены наиболее важные фрагменты программного кода, взятые с небольшими правками из типового алгоритма.

Задача «Алгоритм Евклида для поиска НОД (наибольший общий делитель) двух чисел».

Допустим, даны два числа: **X=48, Y=66**.Алгоритм Евклида заключается в следующем: на каждом шаге большее из чисел заменяется остатком от деления большего на меньшее число. Окончание алгоритма наступает при обнулении одного из чисел; второе, ненулевое, число и будет искомым НОД. В приведённой на рис. 2 иллюстрации алгоритма **НОД(48, 66)=6**.

	Дано:	1 шаг:	2 шаг:	3 шаг:	4 шаг:
y	48	48	12	12	0
x	66	18	18	6	6

Рис. 2. Алгоритм Евклида



А теперь вопрос «на засыпку» – найдите отличия в решениях за-

дач – поиска НОД и разрезания на квадраты ☺!

```
var x,y,z:integer;
begin
  readln (x,y);
  if x<y then begin
    z:=x; x:=y; y:=z;
  end;
  while ((x>0)and(y>0)) do
  begin
    x:=x mod y;
    z:=x; x:=y; y:=z;
  end;
  writeln (x+y);
end.
```

Задача «Перевод чисел из десятичной системы счисления (с.с.) в В-ричную (B<10)».

Правило перевода рассмотрим на примере: $(242)_{10} = (362)_8$ (см. рис. 3):

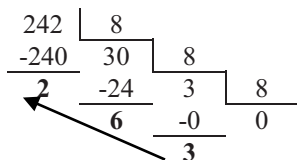


Рис. 3. Иллюстрация алгоритма перевода чисел из 10-тичной с.с. в 8-ричную

Идею решения хорошо иллюстрирует схема выше: число 242 делим на 8, получаем остаток, затем целую часть частного от деления

числа 242 на 8 опять делим на 8. И так до получения нулевого частного.

Проблема заключается в одном – как организовать «сбор» остатков в обратном направлении?

Предлагаемый вариант решения таков: первый остаток – младшая цифра числа, второй остаток – цифра, стоящая в следующем (старшем) разряде, третья цифра – еще более «старшая». Сдвиг цифры по разрядной сетке влево реализуем путём умножения очередной полученной цифры на 1, затем на 10, затем на 100 и т.д. (результат будем складывать). Степени десятки будем получать, увеличивая значение переменной **ST** в 10 раз в теле цикла.

```
var a,b,s,x,st: integer;
begin
  readln (a,b);
  s:=0;
  st:=1;
  while (a>0) do
  begin
    x:=a mod b;
    s:=s+x*st;
    st:=st*10;
    a:=a div b;
  end;
  writeln (s);
end.
```

Рассмотренные выше задачи достаточно интересны – как минимум две из них входят в разряд занимательных задач. Но даже «сухая», с предельной математической точностью, формулировка задачи отнюдь не уменьшает её «занимательности»! Рассмотрим ещё одну группу задач, решение которых базируется на выведенном ранее типовом алгоритме, а условие перекликается с классом задач **«Теория чисел (Разбор числа на цифры)»**.

Для начала оговорим пару деталей:

- использование операций целочисленного деления на 2 позволит определять четность числа (на K – кратность числа числу K).
- использование операций целочисленного деления на 10 позволит оперировать цифрами числа («перемещаться» по разрядной сетке числа).

Задача. Ввести многозначное число. Найти произведение, минимальную, максимальную цифру числа.

Кратко изложим основные моменты реализации задачи:

- Ячейка, в которой накапливается произведение в цикле, должна быть предварительно заполнена единицей

(иначе мы никогда не накопим произведение, так как оно в цикле будет обнуляться – ведь умножение происходит на переменную, значение которой нулевое).

- Переменную, хранящую максимальную цифру числа заполняем нулём (все остальные цифры будут больше либо равны нулю), минимальную – девяткой (все остальные цифры будут меньше либо равны девятке).
- Получение младшей цифры, как уже было разобрано выше, осуществляется путём арифметической операции $x \bmod 10$.
- Как только младшая цифра получена, её необходимо удалить из ячейки (ведь на новом шаге цикла младшей уже должна быть цифра, стоящая в следующем, более старшем разряде). Отбрасывание из ячейки младшей цифры осуществляется так: $x := x \div 10$.
- Отбрасывание младшей цифры будет происходить циклически до тех пор, пока цифры в ячейке «не кончатся» (переменная X не обнулится).

«Костяк» программы выглядит так:

```
var x, cifra, i, p, max, min: integer;
begin
  readln (x);
  ...
  while (x>0) do
    begin
      cifra:=x mod 10;
      ...
      x:=x div 10;
    end;
  writeln (...);
end.
```

Будем считать эту программу типовым алгоритмом «Разбора числа на цифры». При необходимости произвести с цифрами числа какие-либо действия (найти сумму, произведение, максимальную, минимальную цифру числа и т.д.) нам потребуется заменить троеточие на команды, осуществляющие обработку цифр числа (в соответствии с изложенными выше основными моментами реализации задачи).

Задача «Перевод чисел из В-ричной системы счисления ($B < 10$) в десятичную».

```
var X,cifra,s,st:integer;  
begin  
  readln(a,b);  
  st:=1;  
  while (a>0) do  
    begin  
      cifra:=a mod 10;  
      s:=s+cifra*st;  
      st:=st*b;  
      a:=a div 10;  
    end;  
  writeln (S);  
end.
```

А вот теперь наступил момент интриги. Недаром в начале статьи был переброшен мостик между весёлой детской песенкой и реализацией этой песенки на языке программирования Паскаль! Нам пора в обратный путь – досконально разобранные в статье типовые алгоритмы решения разнообразных задач (точнее – один основной и один вытекающий из него алгоритм) помогут вам САМОСТОЯТЕЛЬНО справиться с решением предложенных на ЕГЭ двух достаточно

Правило перевода рассмотрим на примере: $(362)_8 = 3 \cdot 8^2 + 6 \cdot 8^1 + 2 \cdot 8^0 = (242)_{10}$.

Как видно из иллюстрации правила перевода числа в десятичную систему счисления число необходимо «разобрать» на цифры. В этом нам пригодится рассмотренный выше алгоритм «Разбора числа на цифры». Каждая цифра числа умножается на основание системы счисления в определенной степени. Возведение в степень реализуем путём умножения переменной **ST** на основание системы счисления в теле цикла.

традиционных заданий. И не забывайте мурлыкать тихонечко «Мы делили апельсин, много нас, а он один...», хорошее настроение – неременный залог успеха при работе с задачами из банка заданий ЕГЭ. Итак, вперед и с песней!

Задача. Ниже записана программа. Получив на вход число **X**, эта программа печатает два числа: **L** и **M**. Укажите наибольшее из таких чисел **X**, при вводе которых алгоритм печатает сначала **3**, а потом **7**.

```
var x, l, m: integer;  
begin  
  readln(x);  
  l:=0; m:=0;  
  while x>0 do  
    begin  
      l:= l+1;  
      m:= m+(x mod 10);  
      x:= x div 10;  
    end;  
  writeln(l); write(m);  
end.
```

Задача. Требовалось написать программу, при выполнении которой с клавиатуры считывается натуральное число N , не превосходящее 10^9 , и выводится произведение цифр этого числа. Программист торопился и написал программу неправильно (листинг программы приведён ниже). Последовательно выполните следующее:

- Определите, что выведет программа при вводе числа 532.

- Найдите все ошибки в этой программе (их может быть одна или несколько). Для каждой ошибки:
 - выпишите строку, в которой сделана ошибка;
 - укажите, как надо исправить ошибку – приведите правильный вариант строки.

```
var x,cifra,p:integer;  
begin  
  readln(x);  
  p:= x mod 10;  
  while (x>=10) do  
    begin  
      cifra:=x mod 10;  
      p:=p*cifra;  
      x:=x div 10;  
    end;  
  writeln (p);  
end.
```