|  |  |
| --- | --- |
| |  | | --- | | **B8**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;          if x mod 2 = 0 then              M:= M + (x mod 10) div 2;          x:= x div 10;      end;      writeln(L); write(M);  end.**  **Пояснение.** Рассмотрим цикл, число шагов которого зависит от изменения переменной x:  while x > 0 do begin  ...  x:= x div 10;  end;  Т. к. оператор div оставляет только целую часть от деления, то при делении на 10 это равносильно отсечению последней цифры.   Из приведенного цикла видно, что на каждом шаге от десятичной записи *x* отсекается последняя цифра до тех пор, пока все цифры не будут отсечены, то есть x не станет равно 0; поэтому цикл выполняется столько раз, сколько цифр в десятичной записи введенного числа, при этом число L столько же раз увеличивается на 1. Следовательно, конечное значение L совпадает с числом цифр в *x*. Для того, чтобы L стало L=3, *x* должно быть **трёхзначным**.   Теперь рассмотрим оператор изменения M:  if x mod 2 = 0 then    M:= M + (x mod 10) div 2;  end;   Оператор mod оставляет только остаток от деления, при делении на 10 это последняя цифра *x*.  Условие x mod 2 = 0 означает следующее: чтобы M увеличилось, число *x* должно быть чётным.   Предположим, исходное *x* нечётное, тогда на первом шаге M = 0.  Если на втором шаге *x* также нечётное (вторая цифра исходного числа нечётная), то M = 0, причём каким бы ни было значение x на третьем шаге, мы не сможем получить M = 7, поскольку остаток от деления чётного числа на 10 не первосходит 4, а 4 / 2 = 2, следовательно **вторая цифра** исходного *x* **чётная**.   Тогда первая цифра может принимать значения 2, 4, 6, 8, но мы ищем наибольшее *x*, поэтому сделаем первую цифру, равной 9, тогда наше предположение не удовлетворяет условию задачи, и последняя цифра исходного числа обязана быть чётной, т.е. **исходное *x* чётно**.   7 = 4 + 3, чему соответствуют цифры 8 и 6. Теперь, распологая цифры по убыванию, находим наибольшее возможное *x*: *x* = 986.   Ответ: 986.    [Сообщить об ошибке](javascript:void(0))    [Обратиться за помощью](http://inf.reshuege.ru/consultations?prob_id=3209) | |
| |  |  | | --- | --- | |  |  | | B8 | **B8**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 10.   **var x, L, M: integer;  begin      readln(x);      L:=0; M:=0;      while x > 0 do begin          L:=L+1;          if M < x then begin              M:= (x mod 10) \* 2;          end;          x:= x div 10;      end;      writeln(L); write(M);  end.**  **Пояснение.** Рассмотрим цикл, число шагов которого зависит от изменения переменной x:  while x > 0 do begin  ...  x:= x div 10;  end;  Т. к. оператор div оставляет только целую часть от деления, то при делении на 10 это равносильно отсечению последней цифры.   Из приведенного цикла видно, что на каждом шаге от десятичной записи *x* отсекается последняя цифра до тех пор, пока все цифры не будут отсечены, то есть x не станет равно 0; поэтому цикл выполняется столько раз, сколько цифр в десятичной записи введенного числа, при этом число L столько же раз увеличивается на 1. Следовательно, конечное значение L совпадает с числом цифр в *x*. Для того, чтобы L стало L=3, *x* должно быть **трёхзначным**.   Теперь рассмотрим оператор изменения M:  if M < x then begin    M:= (x mod 10) \* 2;  end;   Оператор mod оставляет только остаток от деления, при делении на 10 это последняя цифра *x*.   Чтобы M приняло значение M = 10 в числе x должно присутствовать цифра 5.  Т. к. мы ищем наибольшее *x*, сделаем первую цифру исходного числа, равной 9, при этом если вторая цифра 5, тогда на третьем шаге условие M < *x* не выполняется (10>9), и на экране мы получим нужное нам число.   Осталось определить последнюю цифру исходного числа *x*. После первого шага *x* = 95, но максимальный остаток от деления на 10 равен 9, следовательно, M после первого шага не может превысить 9 \* 2 = 18, а 18 < 95, значит будет выполняться нужный нам второй шаг.   Искомое число 959.   Ответ: 959. | |
|  |