

Решение задачи 10 ОГЭ 2017 по информатике

Задачи типа 10 ОГЭ по информатике относятся к довольно сложным. В подобных задачах проверяется знание учащегося работы с одномерными массивами, а также с циклами.

Рассмотрим решение такой задачи на примере демоверсии ФИПИ по информатике за 2017 год. Напомню, что на сайте уже [есть решение](#) подобной задачи.

Условие:

В таблице Dat представлены данные о количестве голосов, поданных за 10 исполнителей народных песен (Dat[1] – количество голосов, поданных за первого исполнителя; Dat[2] – за второго и т. д.). Определите, какое число будет напечатано в результате работы следующей программы. Текст программы приведён на трёх языках программирования.

Алгоритмический язык	Бейсик	Паскаль
алг нач целтаб Dat[1:10] цел k, m Dat[1] := 16 Dat[2] := 20 Dat[3] := 20 Dat[4] := 41 Dat[5] := 14 Dat[6] := 21 Dat[7] := 28 Dat[8] := 12 Dat[9] := 15 Dat[10] := 35 m := 0 нц для k от 1 до 10 если Dat[k]>m то m := Dat[k] все кц вывод m кон	DIM Dat(10) AS INTEGER DIM k,m AS INTEGER Dat(1) = 16: Dat(2) = 20 Dat(3) = 20: Dat(4) = 41 Dat(5) = 14: Dat(6) = 21 Dat(7) = 28: Dat(8) = 12 Dat(9) = 15: Dat(10) = 35 m = 0 FOR k = 1 TO 10 IF Dat(k)>m THEN m = Dat(k) ENDIF NEXT k PRINT m	Var k, m: integer; Dat: array[1..10] of integer; Begin Dat[1] := 16; Dat[2] := 20; Dat[3] := 20; Dat[4] := 41; Dat[5] := 14; Dat[6] := 21; Dat[7] := 28; Dat[8] := 12; Dat[9] := 15; Dat[10] := 35; m := 0; for k := 1 to 10 do if Dat[k]>m then begin m := Dat[k] end; writeln(m); End.

Решение:

Итак, давайте рассмотрим программу для того, чтобы понять, что, собственно, она делает. Я буду рассматривать программу, написанную на Паскале. Возможно, мое объяснение будет некорректно с точки зрения классического программирования и профессиональной терминологии, но моя задача объяснить, как это работает.

Посмотрим на первые строки:

```
Var k, m: integer;  
Dat: array[1..10] of integer;
```

В них мы говорим Паскалю, что будем использовать две целочисленные переменные — k и m , а также описываем массив, который называем Dat . Этот массив состоит из 10 элементов, в качестве которых выступают целые числа. Далее начинается блок операторов и самое интересное.

Первое, что мы делаем — заполняем описанный массив Dat значениями. Это происходит в следующих строках:

```
Dat[1] := 16; Dat[2] := 20;  
Dat[3] := 20; Dat[4] := 41;  
Dat[5] := 14; Dat[6] := 21;  
Dat[7] := 28; Dat[8] := 12;  
Dat[9] := 15; Dat[10] := 35;
```

Думаю, понятно, что в первый элемент массива мы кладем число 16, во второй 20, в третий 20 и так далее. Это можно отобразить более наглядно в виде таблицы (кликните для увеличения).

Массив Dat										
Индекс	1	2	3	4	5	6	7	8	9	10
Значение	16	20	20	41	14	21	28	12	15	35

Задача 10 ОГЭ по информатике

Итак, массив мы заполнили, дальше пойдет самое интересное. Переменной m мы присваиваем значение, равное нулю. А после этого запускаем цикл:

```
for k := 1 to 10 do  
if Dat[k]>m then  
begin  
  m := Dat[k]  
end;
```

Давайте проанализируем, что происходит в этом цикле. По первой строчке (**for k := 1 to 10 do**) мы видим, что в цикле переменная k меняется от 1 до

10 с шагом 1. Далее условный оператор (**if Dat[k]>m then**), который сравнивает значение текущего элемента массива и значение переменной m. И если элемент массива больше, то в переменную m записываем значение этого элемента. Давайте попробуем пошагово.

Итак, в переменной m у нас ноль, начинаем выполнение цикла. Значение k равно единице. Подставим в условие значения Dat[1] (а это 16) и m и получим:

if 16>0 then

Как видим, это условие истинно, значит выполним следующее присваивание $m := \text{Dat}[1]$ и в переменной m вместо нуля уже окажется число 16. Первый шаг цикла мы отработали. Перейдем ко второму.

На втором шаге переменная k равна 2, а условие примет вид $\text{if Dat}[2]>m$. Подставим значения и получим $\text{if } 20>16 \text{ then}$. Условие опять истинное, значит выполнив следующий оператор присваивания $m := \text{Dat}[k]$ получим в переменной m число 20.

На третьем шаге будем сравнивать значение третьего элемента массива (20) и значение переменной m (20). Это условие будет ложным, так как 20 не больше 20, а значит и оператор присваивания мы не выполняем, переходя к четвертому шагу.

На 4 шаге получим, что $\text{Dat}[4] = 41$, а в переменной m число 20. Так как $41 > 20$, то в m у нас окажется 41.

Этого уже достаточно, чтобы понять логику программы. Подумайте сами, прежде чем прочитать мой ответ. На самом деле, мы проходим по всем элементам массива и сравниваем его наибольшим предыдущим. И если элемент больше, то запоминаем его в переменной m и выводим его на экран (`writeln(m)`). Следовательно, в переменной m в конце выполнения программы у нас будет самый больший элемент массива Dat. А это 41. Это и будет ответом — 41.

Ответ: 41.