

Основы программирования. Массивы (язык Pascal)

Прудаева Ирина Владимировна,

учитель информатики МАОУ гимназии №21 города Тюмени

методист МАОУ «Информационно-методического центра» города
Тюмени

ст. преподаватель кафедры системной и программной инженерии
ИМиКН ТюмГУ

Структура программы

```
program имя программы;  
раздел uses  
раздел описаний  
begin  
    операторы  
end.
```

Раздел описания переменных начинается с ключевого слова **var**, после которого следуют элементы описания вида

список имен: тип;

или

имя: тип := выражение;

К **простым** относятся [целые](#) и [вещественные](#) типы, [логический](#), [символьный](#), [перечислимый](#) и [диапазонный](#) тип.

Тип данных называется **структурированным**, если в одной переменной этого типа может содержаться множество значений.

К структурированным типам относятся [массивы](#), [строки](#), [записи](#), [кортежи](#), [множества](#), [файлы](#) и [классы](#).

Тип	Размер, байт	Диапазон значений
<code>shortint</code>	1	-128..127
<code>smallint</code>	2	-32768..32767
<code>integer</code> , <code>longint</code>	4	-2147483648..2147483647
<code>int64</code>	8	-9223372036854775808..9223372036854775807
<code>byte</code>	1	0..255
<code>word</code>	2	0..65535
<code>longword</code> , <code>cardinal</code>	4	0..4294967295
<code>uint64</code>	8	0..18446744073709551615
<code>BigInteger</code>	переменный	неограниченный

Вещественные типы

Ниже приводится таблица вещественных типов, содержащая их размер, количество значащих цифр и диапазон допустимых значений:

Тип	Размер, байт	Количество значащих цифр	Диапазон значений
<code>real</code>	8	15-16	$-1.8 \cdot 10^{308} .. 1.8 \cdot 10^{308}$
<code>double</code>	8	15-16	$-1.8 \cdot 10^{308} .. 1.8 \cdot 10^{308}$
<code>single</code>	4	7-8	$-3.4 \cdot 10^{38} .. 3.4 \cdot 10^{38}$
<code>decimal</code>	16	28-29	-79228162514264337593543950335 .. 79228162514264337593543950335

Логический тип

Значения логического типа `boolean` занимают 1 байт и принимают одно из двух значений, задаваемых predetermined константами `True` (истина) и `False` (ложь).

Символьный тип

Символьный тип `char` занимает 2 байта и хранит Unicode-символ.

Массивы

Массив представляет собой набор элементов одного типа, каждый из которых имеет свой номер, называемый *индексом* (индексов может быть несколько, тогда массив называется *многомерным*).

- Task1 Заполнить массив четными числами от 2 до $2N$ по порядку
- Task2 Заполнить массив числами от N до 1
- Task3 Заполнить массив по правилу 1,-2,3,-4,5,-6
- Task4 Заполнить массив по правилу 1,0,2,0,3,0,4
- Task5 Вычислить количество четных чисел
- Task6 Вычислить сумму элементов массива кратных их номерам
- Task7 Вычислить сумму элементов массива оба соседа которого положительны
- Task8 Определить максимум
- Task9 Определить есть ли в массиве элементы, равные некоторому числу K
- Task10 Найти позиции первого и последнего элементов массива, равному минимальному
- Task11 Заменить элементы массива минимальным из его соседей
- Task12 Заменить элементы массива максимальным из последующих элементов
- Task13 Заменить элементы массива суммой предшествующих элементов
- Task14 Заменить элементы массива количеством равных ему элементов
- Task15 Заменить числом 0 элементы значения которых заканчиваются цифрой 7

Диагностическая работа от 14.09.17 (Статград)

25

Дан массив, содержащий 2018 положительных целых чисел, не превышающих 10000. Необходимо найти и вывести **максимальный** из тех элементов этого массива, десятичная и шестнадцатеричная запись которых заканчивается одним и тем же символом. Если таких чисел в массиве нет, ответ считается равным нулю.

Например, для массива из четырёх элементов, равных 80, 91, 162 и 250, в ответе должно получиться 162. В шестнадцатеричном виде эти числа записываются как 50, 5B, A2 и FA. Последний символ в десятичной и шестнадцатеричной записи совпадает у первого и третьего числа, большее из них – третье, то есть 162.

Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать часть из описанных.

Паскаль

```
const
    N=2018;
var
    a: array [1..N] of integer;
    i, m, k: integer;
begin
    for i:=1 to N do
        readln(a[i]);
    ...
end.
```

Содержание верного ответа

```
m:=0;  
for i:=1 to n do  
    if (a[i] mod 10=a[i] mod 16) and (a[i]>m) then  
        m:=a[i];  
writeln (m);
```

Указания по оцениванию	Баллы
<p>Предложен правильный алгоритм, выдающий верное значение.</p> <p>Допускается запись алгоритма на другом языке, использующая аналогичные переменные. В случае если язык программирования использует типизированные переменные, описания переменных должны быть аналогичны описаниям переменных на языках, использованных в задании. Использование нетипизированных или необъявленных переменных возможно только в случае, если это допускается языком программирования, при этом количество переменных и их идентификаторы должны соответствовать условию задачи. В алгоритме, записанном на языке программирования, допускается наличие отдельных синтаксических ошибок, не искажающих замысла автора программы</p>	2
<p>Не выполнены условия, позволяющие поставить 2 балла.</p> <p>Предложено в целом верное решение, содержащее не более одной ошибки из числа следующих (если одинаковая ошибка повторяется несколько раз, она считается за одну ошибку).</p> <ol style="list-style-type: none"> 1) Отсутствие инициализации или неверная инициализация максимума. 2) Выход за границы массива. 3) Неверные сравнения при определении максимума (например, вместо максимума ищется минимум). 4) Неверное определение последней цифры (например, использование <code>div</code> вместо <code>mod</code> или аналогичная замена в других языках). 5) Неверный ответ или отсутствие ответа в ситуации, когда в массиве нет подходящих элементов. 	1

6) Ошибка при построении логического выражения. 7) Отсутствует вывод ответа. 8) Используется переменная, не объявленная в разделе описания переменных. 9) Индексная переменная в цикле не меняется (например, в цикле while) или меняется неверно	
Не выполнены условия, позволяющие поставить 1 или 2 балла	0
<i>Максимальный балл</i>	<i>2</i>

Дан набор из N целых положительных чисел. Для каждого числа вычисляется сумма двух последних цифр в его десятичной записи (для однозначных чисел предпоследняя цифра считается равной нулю). Необходимо определить, какая сумма при этом получается чаще всего. Если таких сумм несколько, необходимо вывести наибольшую из них.

Напишите эффективную по времени и по памяти программу для решения этой задачи.

Программа считается эффективной по времени, если при увеличении количества исходных чисел N в k раз время работы программы увеличивается не более чем в k раз.

Программа считается эффективной по памяти, если память, необходимая для хранения всех переменных программы, не превышает 1 Кбайт и не увеличивается с ростом N .

Максимальная оценка за правильную (не содержащую синтаксических ошибок и дающую правильный ответ при любых допустимых входных данных) программу, эффективную по времени и по памяти, – 4 балла.

Максимальная оценка за правильную программу, эффективную только по времени или только по памяти, – 3 балла.

Максимальная оценка за правильную программу, не удовлетворяющую требованиям эффективности, – 2 балла.

Вы можете сдать одну или две программы решения задачи. Если Вы сдадите две программы, каждая из них будет оцениваться независимо от другой, итоговой станет бóльшая из двух оценок.

Перед текстом программы кратко опишите алгоритм решения. Укажите использованный язык программирования и его версию.

Описание входных и выходных данных

В первой строке входных данных задаётся количество чисел N ($1 \leq N \leq 1000$).

В каждой из последующих N строк записано одно натуральное число, не превышающее 10000.

Пример входных данных:

5
15
417
123
6
4841

Пример выходных данных для приведённого выше примера входных данных:

6
Суммы двух последних цифр для чисел из данного набора равны 6, 8, 5, 6, 5.
Чаще других (по два раза) встречаются 6 и 5, в ответе выводится бóльшая из этих сумм.

Пример правильной и эффективной программы на языке Паскаль

```
var
  N: integer;      {количество чисел}
  a: integer;      {очередное число}
  s: integer;      {сумма двух последних цифр}
  d: array [0..18] of integer; {подсчёт сумм}
  mx: integer;     {максимальное количество сумм}
  imx: integer;    {самая частая сумма}
  i: integer;

begin
  for i:=0 to 18 do d[i]:=0;
  readln(N);
  for i:=1 to N do begin
    readln(a);
    s := a mod 10 + a div 10 mod 10;
    d[s] := d[s]+1;
  end;
  mx := 0;
  for i:=0 to 18 do begin
    if d[i] >= mx then begin
      mx := d[i];
      imx := i;
    end
  end;
  writeln(imx)
end.
```

Указания по оцениванию	Баллы
<p>Если в работе представлены две программы решения задачи, то каждая из них независимо оценивается по указанным ниже критериям, итоговой считается бóльшая из двух оценок</p>	
<p>Программа правильно работает для любых входных данных произвольного размера. Используемая память не зависит от количества прочитанных чисел, а время работы пропорционально этому количеству.</p> <p>Допускается наличие в тексте программы до трёх синтаксических ошибок одного из следующих видов:</p> <ol style="list-style-type: none">1) пропущен или неверно указан знак пунктуации;2) неверно написано или пропущено зарезервированное слово языка программирования;3) не описана или неверно описана переменная;4) применяется операция, недопустимая для соответствующего типа данных. <p>Если одна и та же ошибка встречается несколько раз, это считается за одну ошибку</p>	4

Не выполнены условия, позволяющие поставить 4 балла.

3

Программа в целом работает правильно для любых входных данных произвольного размера. Время работы пропорционально количеству введённых чисел, правильно указано, какие величины должны вычисляться по ходу чтения элементов последовательности чисел.

Используемая память, возможно, зависит от количества прочитанных чисел (например, входные данные запоминаются в массиве, контейнере STL в C++ или другой аналогичной структуре данных).

Количество синтаксических ошибок (описок), указанных в критериях на 4 балла, не более пяти.

Допускается наличие не более одной ошибки следующих видов:

- 1) ошибка при инициализации или отсутствие инициализации счётчиков;
- 2) использование нахождения остатка вместо деления (mod вместо div в Паскале) или наоборот;
- 3) ошибка при построении цикла разбиения на цифры (если в программе используется такой цикл);
- 4) отсутствие инициализации или неверная инициализация максимума;
- 5) допущен выход за границу массива;
- 6) вместо максимума ищется минимум;
- 7) при наличии нескольких ответов выводится не максимальный из них;
- 8) вместо суммы (индекса массива счётчиков) выводится количество (значение элемента)

<p>Не выполнены условия, позволяющие поставить 3 или 4 балла.</p> <p>Программа работает в целом верно, эффективно или нет, но в реализации алгоритма есть до трёх содержательных ошибок, описанных в критериях на 3 балла.</p> <p>Количество синтаксических ошибок, указанных в критериях на 4 балла, не должно быть более девяти</p>	2
<p>Не выполнены условия, позволяющие поставить 2, 3 или 4 балла.</p> <p>Программа работает правильно в отдельных частных случаях.</p> <p>Допускается любое количество синтаксических ошибок</p>	1
<p>Не выполнены условия, позволяющие поставить 1, 2, 3 или 4 балла</p>	0
	<i>Максимальный балл</i>
	4