

25 (C2) (высокий уровень, время – 30 мин)

Тема: Обработка массива (написать программу из 10-15 строк на языке программирования или алгоритм на естественном языке).

Что нужно знать:

- *массив* – это набор однотипных элементов, имеющих общее имя и расположенных в памяти рядом
- для обращения к элементу массива используют квадратные скобки, запись **A[i]** обозначает элемент массива **A** с номером (индексом) **i**
- для обработки всех элементов массива используется цикл вида¹

```
for i:=1 to N do begin
  { что-то делаем с элементом A[i] }
end;
```

переменная **i** обозначает номер текущего элемента массива, она меняется от 1 до N с шагом 1, то есть мы “проходим” последовательно все элементы
- *матрица* (двухмерный массив) – это прямоугольная таблица однотипных элементов
- если матрица имеет имя **A**, то обращение **A[i, k]** обозначает элемент, расположенный на пересечении строки **i** и столбца **k**

		k	
i		A[i, k]	

- каждая строка матрицы – это обычный (одномерный, линейный) массив; для того, чтобы обработать строку **i** в матрице из **M** столбцов, нужно использовать цикл, в котором меняется номер столбца **k**:

```
for k:=1 to M do begin
  { что-то делаем с элементом A[i, k] }
end;
```
- каждый столбец матрицы – это обычный (одномерный, линейный) массив; для того, чтобы обработать столбец **k** в матрице из **N** строк, нужно использовать цикл, в котором изменяется номер строки **i**:

```
for i:=1 to N do begin
  { что-то делаем с элементом A[i, k] }
end;
```
- условие задачи записано на нескольких языках (алгоритмический язык, Паскаль, Бейсик и Си); в принципе, решение можно писать и на любом другом языке, в том числе на естественном языке или в виде блок-схемы.

При решении задачи 25 ЗАПРЕЩАЕТСЯ использовать библиотечные подпрограммы языка программирования, например, подпрограммы сортировки и т.п.

¹ По традиции нумерация элементов массива в Паскале обычно начинается с единицы, далее N обозначает размер массива (количество элементов).

Пример задания:

P-04. Дан целочисленный массив из 20 элементов. Элементы массива могут принимать целые значения от $-10\,000$ до $10\,000$ включительно. Опишите на естественном языке или на одном из языков программирования алгоритм, позволяющий найти и вывести количество пар элементов массива, сумма которых нечётна и положительна. Под парой подразумевается два подряд идущих элемента массива.

Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но использовать все описанные переменные не обязательно.

Паскаль	Алгоритмический язык
<pre>const N = 20; var a: array [1..N] of integer; i, j, k: integer; begin for i := 1 to N do readln(a[i]); ... end.</pre>	<pre>алг нач цел N = 20 цел таб a[1:N] цел i, j, k нц для i от 1 до N ввод a[i] кц ... кон</pre>

Решение:

- даже если вы хорошо владеете программированием, сначала лучше (прежде всего, для себя) написать алгоритм на русском языке
- в задании нужно перебрать все пары соседних элементов, начиная от пары (**a[1]**, **a[2]**) до пары (**a[N-1]**, **a[N]**) и подсчитать, для скольких пар сумма элементов нечётна и положительна
- поскольку нам нужно что-то считать, придётся использовать переменную счётчик; сначала её значение равно 0, затем, при каждой найденной подходящей паре, значение счётчика увеличивается на 1; в качестве счётчика можно использовать любую целую переменную из тех, что были объявлены в условии, например, **k**:

```
k := 0;
перебрать все пары
  если пара подходящая, то
    k := k + 1
```
- как же перебрать все пары? примем, что переменная **i** будет хранить номер первого элемента в паре, то есть, будем рассматривать пары (**a[i]**, **a[i+1]**)
- очевидно, что нужно организовать цикл, который перебирает все значения **i** в интервале от 1 (для первой пары, (**a[1]**, **a[2]**)) до N-1 (для последней пары, (**a[N-1]**, **a[N]**))

```
for i:=1 to N-1 do ...
```
- как определить, что пара (**a[i]**, **a[i+1]**) подходящая? нужно вычислить её сумму **a[i]+a[i+1]** и проверить, верно ли, что эта сумма нечётна и положительна
- нечётность проверим, вычислив остаток от деления на 2; если этот остаток не равен 0, число нечётное
- получается такой цикл, после которого нужно не забыть вывести результат – значение счётчика **k**:

```
k := 0;
for i:=1 to N-1 do
  if (a[i]+a[i+1] > 0) and ((a[i]+a[i+1]) mod 2 <> 0) then
    k := k + 1;
writeln(k);
```

- 9) для того, чтобы не вычислять дважды сумму в условном операторе, можно было записать её в свободную переменную j:

```
k := 0;
for i:=1 to N-1 do begin
  j := a[i]+a[i+1];
  if (j > 0) and (j mod 2 <> 0) then
    k := k + 1
end;
writeln(k);
```

- 10) возможен и ответ на естественном языке (русском):

Записать значение 0 в переменную k. Затем в цикле перебрать все пары соседних элементов, начиная с пары (a[1], a[2]) до пары (a[N-1], a[N]). Для каждой пары вычислить сумму, если эта сумма положительна и нечётна (остаток от её деления на 2 не равен нулю), увеличить значение k на единицу. После завершения цикла вывести значение k.

Возможные проблемы:

- не забудьте сказать, что нужно вывести после окончания работы программы
- не забудьте задать правильное начальное значение для счётчика
- не забудьте, что в Паскале каждое из простых условий (отношений) нужно брать в скобки
- не забудьте, что если в теле цикла нужно выполнить несколько операторов, их нужно заключать в "операторные скобки" **begin-end**
- цикл по i должен заканчиваться не на N, а на N-1, чтобы не было выхода за границы массива (рассматриваются элементы a[i] и a[i+1])
- если вы достаточно хорошо владеете русским языком для того, чтобы понятно излагать свои мысли, с точки зрения тактики рекомендуется писать алгоритм на русском языке – по крайней мере, тут не снизят балл за пропущенную точку с запятой
- посмотрите внимательно диапазон, в котором находятся исходные числа; дело в том, что во многих языках, например, в Паскале и в Си, остаток от деления отрицательного числа на положительное – число отрицательное, например $(-7) \bmod 2 = -1$, поэтому определять, например, "неделимость" элемента массива на 2 с помощью условия $a[i] \bmod 2 = 1$ нельзя (не будет работать для отрицательных чисел), нужно использовать условие $a[i] \bmod 2 <> 0$.

Ещё пример задания:

P-03. Дан целочисленный массив из 20 элементов. Элементы массива могут принимать целые значения от 0 до 1000. Опишите на русском языке или на одном из языков программирования алгоритм, позволяющий найти и вывести минимальное значение среди элементов массива, которые имеют чётное значение и не делятся на три. Гарантируется, что в исходном массиве есть хотя бы один элемент, значение которого чётно и не кратно трем.

Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но использовать все описанные переменные не обязательно.

Естественный язык:

Объявляем массив A из 20 элементов.

Объявляем целочисленные переменные I, J, MIN.

В цикле от 1 до 20 вводим элементы массива A с 1-го по 20-й.

Паскаль:

```
const N=20;
```

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

Решение:

- даже если вы хорошо владеете программированием, сначала лучше (прежде всего, для себя) написать алгоритм на русском языке
- здесь требуется найти минимальный элемент из всех, которые имеют чётное значение и не делятся на 3
- делимость одного целого числа на другое проверяется с помощью операции взятия остатка (в Паскале – операция **mod**): первое число делится на второе, если остаток от деления равен 0
- тогда условие, определяющее отбор нужных элементов, запишется в виде $(a[i] \bmod 2 = 0) \text{ and } (a[i] \bmod 3 <> 0)$
- стандартный цикл поиска минимального элемента, удовлетворяющего условию, выглядит так:

```
for i:=1 to N do
  if <условие верно> and (a[i] < min) then
    min := a[i];
```

- остается один вопрос: каким должно быть начальное значение переменной **min**? его нужно выбрать таким, чтобы для первого же "подходящего" элемента выполнялось условие $a[i] < \text{min}$, и это "временное" начальное значение было бы заменено на реальное
- к счастью, диапазон входных чисел ограничен (от 0 до 1000), поэтому можно выбрать любое значение, больше 1000, например, 1001 или 9999²
- таким образом, решение задачи на естественном языке выглядит так:
Записываем в переменную **min** значение 1001.
Затем в цикле просматриваем все элементы массива, с первого до последнего. Если остаток от деления очередного элемента на 2 равен 0 и остаток от его деления на 3 не равен нулю и значение элемента меньше, чем значение переменной **min**, записать в переменную **min** значение рассматриваемого элемента массива. Затем переходим к следующему элементу.
После окончания работы цикла выводим значение переменной **min**.

- 9) аналогичное решение на Паскале:

```
min:=1001;
for i:=1 to N do
  if (a[i] mod 2=0) and (a[i] mod 3 <> 0) and (a[i]<min) then
    min:=a[i];
writeln(min);
```

Возможные проблемы:

- не забудьте сказать, что нужно вывести после окончания работы программы
- если вы достаточно хорошо владеете русским языком для того, чтобы понятно излагать свои мысли, с точки зрения тактики рекомендуется писать алгоритм на русском языке – по крайней мере, тут не снизят за пропущенную точку с запятой
- посмотрите внимательно диапазон, в котором находятся исходные числа; дело в том, что во многих языках, например, в Паскале и в Си, остаток от деления отрицательного числа на положительное – число отрицательное, например $(-7) \bmod 3 = -1$, поэтому определять,

² Вообще говоря, в данной задаче не требуется находить номер минимального элемента, поэтому сначала можно записать в переменную **min** число 1000 – проверьте, что программа все равно выдаст верное значение.

например, "неделимость" элемента массива на 3 с помощью условия $a[i] \bmod 3 = 1$ нельзя (не будет работать для отрицательных чисел), нужно использовать условие $a[i] \bmod 3 < 0$.

Ещё пример задания:

P-02. Опишите на русском языке или одном из языков программирования алгоритм получения из заданного целочисленного массива размером 30 элементов другого массива, который будет содержать модули значений элементов первого массива (не используя специальной функции, вычисляющей модуль числа).

Решение:

- даже если вы хорошо владеете программированием, сначала лучше (прежде всего, для себя) написать алгоритм на русском языке (или на псевдокоде – это нечто среднее между словесным алгоритмом и готовой программой)
- по условию нужно выделить в памяти два массива одинакового размера, назовем их **A** и **B**; обозначим размер массивов через **N**, индексы элементов изменяются от 1 до **N**;
- в цикле в каждый элемент **B[i]** массива **B** нужно записать модуль соответствующего элемента **A[i]** массива **A**, это нужно сделать для всех **i** от 1 до **N**
- есть небольшая сложность: запрещено использовать стандартную функцию вычисления модуля; согласно определению модуля решение может быть такое: если элемент **A[i]** больше или равен нулю, записываем в **B[i]** его значение без изменений, а если меньше нуля – меняем знак, то есть, в **B[i]** записываем $-A[i]$
- решение в виде алгоритма на русском языке может выглядеть так:
"Выделяем в памяти второй массив того же размера. В цикле рассматриваем все элементы первого массива с первого до последнего. Если текущий элемент больше нуля или равен нулю, в соответствующий элемент второго массива записываем его значение без изменений; если текущий элемент меньше нуля, во второй массив записываем значение элемента с обратным знаком. Выводим второй массив на экран".

- осталось написать программу, практически дословно реализующую это решение:

```
const N=30;
var a, b:array[1..N] of integer;
i: integer;
begin
for i:=1 to N do { ввод всех элементов массива с клавиатуры }
  read(a[i]);
for i:=1 to N do { формирование массива B }
  if a[i]<0 then
    b[i]:=-a[i]
  else b[i]:=a[i];
writeln('Результат:');
for i:=1 to N do { вывод всех элементов массива B }
  write(b[i], ' ');
end.
```

- размер массива грамотно задавать через константу (`const N=30;`), а не вписывать число в каждый цикл; тогда, если нужно будет переделать программу для массива другого размера, достаточно будет изменить всего одно число в начале программы

Возможные проблемы:

- проверяйте правильность минимального и максимального значения переменной цикла в заголовке цикла `for`

- не забывайте вывести результат в конце работы программы

Еще пример задания:

P-01. Опишите на русском языке или одном из языков программирования алгоритм подсчета максимального количества подряд идущих совпадающих элементов в целочисленном массиве длины 30.

Решение:

- сначала нужно понять задачу; предположим, что в массиве есть одинаковые элементы, стоящие рядом:

1	1	2	2	1	1	1	1	3	3	2	2
---	---	---	---	---	---	---	---	---	---	---	---

- самая длинная цепочка стоящих рядом элементов в данном случае состоит из 4-х единиц (она выделена желтым фоном)
- нам нужно по крайней мере две переменных: для хранения номера текущего элемента (при обработке массива в цикле) и для хранения максимального количества идущих подряд элементов (обозначим ее **kMax**)
- в целом (пока неточный) алгоритм может выглядеть так: "пройти весь массив, подсчитывая для каждого элемента длину цепочки подряд идущих одинаковых чисел, если эта длина больше **kMax**, то записать ее в **kMax**"
- отсюда сразу следует, что необходима еще одна переменная (обозначим ее через **k**), пока-зывающая для каждого элемента массива длину цепочки одинаковых чисел, которая заканчивается на этом элементе:

	1	1	2	2	1	1	1	1	3	3	2	2
k	1	2	1	2	1	2	3	4	1	2	1	2
kMax	1	2	2	2	2	2	3	4	4	4	4	4

- следующий шаг к решению: нужно понять, как изменять переменную **k** при проходе по массиву; можно сделать так: если очередной элемент равен предыдущему, счетчик **k** увеличиваем на единицу, а если не равен – записываем в него 1 (цепочка одинаковых чисел кончилась, началась новая, в ней пока один элемент)
- при таком подходе проблема может возникнуть при просмотре первого элемента, потому что для него нет предыдущего; поэтому описанную выше процедуру будем в цикле применять ко всем элементам массива, начиная **со второго** (а не с первого); в самом начале программы запишем в **k** и **kMax** по единице – таким образом, мы "вручную" (без цикла) рассмотрели первый элемент массива
- теперь можно написать алгоритм на русском языке:
"Выделим две вспомогательные переменные, **k** и **kMax**, и запишем в каждую из них по единице. В цикле рассматриваем все элементы массива **со второго** до последнего, если очередной элемент равен предыдущему, увеличиваем **k**; если **k > kMax**, записываем в **kMax** значение **k**. В конце цикла в **kMax** окажется требуемое значение".
- этот алгоритм реализуется в такой программе:

```
const N=30;
var a: array[1..N] of integer;
i, k, kMax: integer;
begin
for i:=1 to N do readln(A[i]); { ввод массива }
k := 1; { обрабатываем A[1] }
kMax := 1;
for i:=2 to N do begin { а теперь в цикле A[2]...A[N] }
  if A[i] = A[i-1] then { цепочка продолжается }
    k := k + 1
```

```

else k := 1;           { цепочка закончилась }
if k > kMax then kMax := k;
end;
writeln(kMax);
end.

```

Возможные проблемы:

- как видим, основная сложность в этой задаче – не написать программу, а придумать хороший (часто еще нужно – быстрый) алгоритм
- проверьте, что будет записано в переменные до начала цикла (определены ли их начальные значения)
- проверяйте, не выйдет ли индекс за границу массива в начале или в конце цикла
- будьте внимательны с "крайними" случаями, например, нужно обязательно убедиться, что программа работает, когда интересующая нас цепочка стоит в самом начале или в самом конце массива

Еще пример задания:

P-00. Дан целочисленный квадратный массив 10×10 . Опишите на русском языке или на одном из языков программирования алгоритм вычисления суммы максимальных элементов из каждой строки. Напечатать значение этой суммы. Предполагается, что в каждой строке такой элемент единственный.

Решение:

- 1) суть задачи: среди элементов каждой строки нужно выбрать максимальный, и все эти выбранные значения сложить
- 2) несложно сразу написать алгоритм на русском языке:
"Чтобы накапливать сумму, нужно ввести целую переменную **Sum**, в которую в самом начале записываем 0; далее в цикле просматриваем все строки, для каждой строки находим максимальный элемент и прибавляем его значение к **Sum**. Для определения максимального элемента в строке вводим переменную **max** и сначала записываем в нее значение первого элемента этой строки. Затем в цикле просматриваем все остальные элементы, начиная со **второго** до конца массива. Если очередной элемент больше значения **max**, записываем в **max** значение этого элемента".
- 3) сначала напишем программу на псевдокоде:

```

const N=10;
{ ввод матрицы N на N }
Sum := 0;
for i:=1 to N do begin
  { max := максимальный элемент в i-ой строке }
  Sum := Sum + max;
end;

```
- 4) остается записать на Паскале те части, которые взяты в фигурные скобки, и до конца оформить программу; по правилам ЕГЭ можно не писать в программе команды для ввода массива, поэтому мы оставим на этом месте комментарий:

```

const N=10;
var A: array[1..N,1..N] of integer;
    i, k, max, Sum: integer;
begin
  { ввод матрицы N на N }

```

```

Sum := 0;
for i:=1 to N do begin
  max := A[i,1];
  for k:=2 to N do
    if A[i,k] > max then max := A[i,k];
  Sum := Sum + max;
end;
writeln(Sum);
end.

```

Возможные проблемы:

- проверьте, правильно ли заданы (и заданы ли вообще) начальные значения для всех переменных
- проверьте, правильно ли расставлены операторные скобки **begin-end**, ограничивающие тело цикла; их обязательно нужно ставить, если в теле цикла несколько операторов
- проверяйте, не выйдет ли индекс за границу массива в начале или в конце цикла
- не перепутайте номер строки (это первый индекс) и номер столбца (второй индекс)
- для надежности не рекомендуется использовать в одной программе переменные **i** и **j**, потому что они слишком похоже выглядят, вот пример ошибочного решения в этой задаче:

```

for i:=1 to N do begin
  max := A[i,1];
  for j:=2 to N do
    if A[j,i] > max then max := A[i,j];
  Sum := Sum + max;
end;

```

- если вы все же используете переменные **i** и **j**, нужно писать их очень четко, чтобы они отличались друг от друга

Немного тактики:

- в этом задании можно написать алгоритм на русском языке, а можно (вместо этого) написать компьютерную программу на одном из языков программирования
- если вы хорошо умеете выражать свои мысли по-русски, **себяпроведы** эксперты рекомендуют писать **только** алгоритм на русском языке; дело в том, что если вы сделаете много ошибок в программе, оценка будет снижена даже при абсолютно правильном алгоритме
- если вам сложно изъясняться на родном языке, а легче записать свои мысли на Паскале или Си – пишите программу, но тщательно проверяйте ее на предмет возможных случайных ошибок-опечаток, которые можно сделать просто по невнимательности:
 - задавайте все начальные значения для переменных
 - проверяйте правильность написания ключевых слов
 - если в теле цикла несколько операторов, заключайте их в блок **begin-end** (операторные скобки)
 - проверяйте начальное и конечное значение переменной цикла
 - если вы используете циклы **while** или **repeat**, проверьте, что переменная цикла изменится в теле цикла (иначе в программе будет закливание, а на ЕГЭ – потерянные баллы)
 - выводите на экран именно то, что требуется по условию
 - ставьте точку с запятой в конце операторов
 - НЕ ставьте точку с запятой перед **else** (Паскаль)

- ставьте точку в конце последнего оператора **end** (Паскаль)

За что снимают баллы:

- задано неверное начальное значение переменных (или вообще не задано)
- неверно указано условие завершения цикла
- "забыли" изменять переменную цикла в цикле **while** (**repeat**)
- перепутаны знаки **<** и **>**, логические операции **or** и **and**
- неверно расставлены операторные скобки **begin-end**
- программа не выводит результат или выводит не то, что спрашивают
- синтаксические ошибки (знаки пунктуации – запятые, точки, точки с запятой; неверное написание ключевых слов) допускаются в разумных пределах (если они не искажают замысел автора)

Задачи для тренировки³:

- 1) Опишите на русском языке или одном из языков программирования алгоритм поиска номера первого из двух последовательных элементов в целочисленном массиве из 30 элементов, сумма которых максимальна (если таких пар несколько, то можно выбрать любую из них).
- 2) Опишите на русском языке или одном из языков программирования алгоритм подсчета числа элементов, равных максимальному, в числовом массиве из 30 элементов.
- 3) Опишите на русском языке или одном из языков программирования алгоритм поиска трех последовательных элементов, сумма которых максимальна, в числовом массиве из 30 элементов.
- 4) Дан прямоугольный массив целых положительных чисел 10 x 20. Опишите на русском языке или на одном из языков программирования алгоритм поиска строки с наименьшей суммой элементов. Вывести на печать номер строки и сумму ее элементов. Предполагается, что такая строка единственна.
- 5) Дан квадратный массив целых положительных чисел 10 x 10. Опишите на русском языке или на одном из языков программирования алгоритм вычисления суммы элементов строки, в которой расположен элемент с максимальным значением. Вывести значение суммы на печать. Предполагается, что такой элемент единственный.
- 6) Дан целочисленный прямоугольный массив 6x10. Опишите на русском языке или на одном из языков программирования алгоритм вычисления суммы минимальных элементов из каждой строки. Напечатать значение этой суммы. Предполагается, что в каждой строке минимальный элемент единственный.
- 7) Дан целочисленный массив из 30 элементов. Элементы могут принимать значения от 0 до 100 – баллы, полученные на ЕГЭ. Опишите на русском языке или на одном из языков программирования алгоритм, который определяет и выводит средний балл учащихся, сдавших экзамен (получивших оценку более 20 баллов). Гарантируется, что хотя бы один ученик в классе успешно сдал экзамен. Исходные данные объявлены так, как показано ниже. Использовать другие переменные запрещается.

Паскаль	Естественный язык
<pre>const N = 30; var A: array[1..N] of integer; i, x, y: integer; s: real; begin for i:=1 to N do readln(A[i]);</pre>	<p>Объявляется массив A из 30 элементов.</p> <p>Объявляются целочисленные переменные i, x и y, а также вещественная переменная s.</p> <p>В цикле от 1 до 30 заполняются все элементы массива A с 1-ого по 30-ый.</p>

³ Источники заданий:

1. Демонстрационные варианты ЕГЭ 2004-2013 гг.
2. Тренировочные работы МИОО.
3. Гусева И.Ю. ЕГЭ. Информатика: раздаточный материал тренировочных тестов. — СПб: Тригон, 2009.
4. Самылкина Н.Н., Островская Е.М. Информатика: тренировочные задания. — М.: Эксмо, 2009.
5. Якушкин П.А., Ушаков Д.М. Самое полное издание типовых вариантов реальных заданий ЕГЭ 2010. Информатика. — М.: Астрель, 2009.
6. Крылов С.С., Ушаков Д.М. ЕГЭ 2015. Информатика. Тематические тестовые задания. — М.: Экзамен, 2015.
7. Ушаков Д.М. ЕГЭ-2015. Информатика. 20 типовых вариантов экзаменационных работ для подготовки к ЕГЭ. — М.: Астрель, 2014.

... end.	...
Си	
<pre>#include<stdio.h> int main(void) { const int N=30; int A[N]; int i, x, y; float s; for (i=0; i<N; i++) scanf("%d", &A[i]); ... }</pre>	

- 8) Дан целочисленный массив из 30 элементов. Элементы могут принимать значения от 160 до 200 – рост учащихся выпускного класса. В баскетбольную команду берут тех, чей рост не менее 180 см. Опишите на русском языке или на одном из языков программирования алгоритм, который определяет и выводит минимальный рост игрока баскетбольной команды. Гарантируется, что хотя бы один ученик играет в баскетбольной команде. Исходные данные объявлены так, как показано ниже. Использовать другие переменные запрещается.

Паскаль	Естественный язык
<pre>const N = 30; var A: array[1..N] of integer; i, x, y: integer; s: real; begin for i:=1 to N do readln(A[i]); ... end.</pre>	<p>Объявляется массив A из 30 элементов.</p> <p>Объявляются целочисленные переменные i, x и y, а также вещественная переменная s.</p> <p>В цикле от 1 до 30 заполняются все элементы массива A с 1-ого по 30-ый.</p> <p>...</p>
Си	
<pre>#include<stdio.h> int main(void) { const int N=30; int A[N]; int i, x, y; float s; for (i=0; i<N; i++) scanf("%d", &A[i]); ... }</pre>	

- 9) Дан целочисленный массив из 31 элемента, в котором записаны значения температуры воздуха в марте. Элементы массива могут принимать значения от (–20) до 20. Опишите на русском языке или на одном из языков программирования алгоритм, который подсчитывает и выводит среднюю температуру по всем дням, когда была оттепель (температура поднималась выше нуля). Гарантируется, что хотя бы один день в марте была оттепель. Исходные данные объявлены так, как показано ниже. Использовать другие переменные запрещается.

Паскаль	Естественный язык
<pre>const N = 31; var A: array[1..N] of integer; i, x, y: integer; s: real; begin for i:=1 to N do readln(A[i]); ... end.</pre>	<p>Объявляется массив A из 31 элемента.</p> <p>Объявляются целочисленные переменные i, x и y, а также вещественная переменная s.</p> <p>В цикле от 1 до 31 заполняются все элементы массива A с 1-ого по 31-ый.</p> <p>...</p>

end.	
Си	
<pre>#include<stdio.h> int main(void) { const int N=31; int A[N]; int i, x, y; float s; for (i=0; i<N; i++) scanf("%d", &A[i]); ... }</pre>	

- 10) Дан целочисленный массив из 30 элементов. Элементы массива могут принимать целые значения от 0 до 100 – баллы учащихся выпускного класса за итоговый тест по информатике. Для получения положительной оценки за тест требовалось набрать не менее 20 баллов. Опишите на русском языке или на одном из языков программирования алгоритм, который находит и выводит минимальный балл среди учащихся, получивших за тест положительную оценку. Известно, что в классе хотя бы один учащийся получил за тест положительную оценку. Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать часть из них.

Паскаль	Естественный язык
<pre>const N=30; var a: array [1..N] of integer; i, j, min: integer; begin for i:=1 to N do readln(a[i]); ... end.</pre>	<p>Объявляется массив A из 30 элементов.</p> <p>Объявляются целочисленные переменные i, j и min.</p> <p>В цикле от 1 до 30 заполняются все элементы массива A с 1-ого по 30-ый.</p> <p>...</p>
Си	
<pre>#include<stdio.h> int main(void) { const int N=30; int a[N]; int i, j, min; for (i=0; i<N; i++) scanf("%d", &a[i]); ... }</pre>	

В качестве ответа Вам необходимо привести фрагмент программы (или описание алгоритма на естественном языке), который должен находиться на месте многоточия. Вы можете записать решение также на другом языке программирования (укажите название и используемую версию языка программирования, например, *Borland Pascal 7.0*) или в виде блок-схемы. В этом случае вы должны использовать те же самые исходные данные и переменные, какие были предложены в условии (например, в образце, записанном на естественном языке).

- 11) Опишите на русском языке или на одном из языков программирования алгоритм определения количества положительных элементов квадратной матрицы, превышающих по величине среднее арифметическое всех элементов главной диагонали.
- 12) Дан целочисленный массив из 30 элементов. Элементы массива могут принимать целые значения от 0 до 100 – баллы учащихся выпускного класса, полученные на экзамене по информатике. Опишите на русском языке или на одном из языков программирования алгоритм, который позволяет найти и вывести количество учащихся, чьи баллы на экзамене выше среднего балла по классу. Ис-

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

Паскаль	Естественный язык
<pre>const N=30; var a: array [1..N] of integer; i, j: integer; s: real; begin for i:=1 to N do readln(a[i]); ... end.</pre>	<p>Объявляем массив A из 30 элементов. Объявляем целочисленные переменные i, j. Объявляем действительную переменную s. В цикле от 1 до 30 вводим элементы массива A с 1-го по 30-й.</p> <p>...</p>
Си	
<pre>#include<stdio.h> int main(void) { const int N=30; int a[N]; int i, j; float s; for (i=0; i<N; i++) scanf("%d", &a[i]); ... }</pre>	

В качестве ответа вам необходимо привести фрагмент программы (или описание алгоритма на естественном языке), который должен находиться на месте многоточия. Вы можете записать решение также на другом языке программирования (укажите название и используемую версию языка программирования, например, *Borland Pascal 7.0*) или в виде блок-схемы. В этом случае вы должны использовать те же самые исходные данные и переменные, какие были предложены в условии (например, в образце, записанном на естественном языке).

- 13) Дан целочисленный массив из 30 элементов. Элементы массива могут принимать произвольные целые значения. С клавиатуры вводится целое число **X**. Опишите на русском языке или на одном из языков программирования алгоритм, который находит и выводит наименьший номер элемента, равного **X**, или сообщение, что такого элемента нет. Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать часть из них.

Паскаль	Естественный язык
<pre>const N=30; var a: array [1..N] of integer; i, j, x: integer; begin for i:=1 to N do readln(a[i]); readln(x); ... end.</pre>	<p>Объявляем массив A из 30 элементов. Объявляем целочисленные переменные i, j, x. В цикле от 1 до 30 вводим элементы массива A с 1-го по 30-й. Вводим переменную X.</p> <p>...</p>
Си	
<pre>#include<stdio.h> int main(void) { const int N=30; int a[N]; int i, j, x; for (i=0; i<N; i++) scanf("%d", &a[i]); scanf("%d", &x); ... }</pre>	

В качестве ответа вам необходимо привести фрагмент программы (или описание алгоритма на естественном языке), который должен находиться на месте многоточия. Вы можете записать решение также на другом языке программирования (укажите название и используемую версию языка программирования, например, *Borland Pascal 7.0*) или в виде блок-схемы. В этом случае вы должны использовать те же самые исходные данные и переменные, какие были предложены в условии (например, в образце, записанном на естественном языке).

- 14) Дан целочисленный массив из 30 элементов. Элементы массива могут принимать произвольные целые значения. Опишите на русском языке или на одном из языков программирования алгоритм, который находит и выводит второй максимум массива (элемент, который в отсортированном по невозрастанию массиве стоял бы вторым). Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать часть из них.

Паскаль	Естественный язык
<pre>const N=30; var a: array [1..N] of integer; i, k, max, max2: integer; begin for i:=1 to N do readln(a[i]); ... end.</pre>	<p>Объявляем массив A из 30 элементов. Объявляем целочисленные переменные i, k, max, max2. В цикле от 1 до 30 вводим элементы массива A с 1-го по 30-й.</p> <p>...</p>
Си	
<pre>#include<stdio.h> int main(void) { const int N=30; int a[N]; int i, k, max, max2; for (i=0; i<N; i++) scanf("%d", &a[i]); ... }</pre>	

В качестве ответа вам необходимо привести фрагмент программы (или описание алгоритма на естественном языке), который должен находиться на месте многоточия. Вы можете записать решение также на другом языке программирования (укажите название и используемую версию языка программирования, например, *Borland Pascal 7.0*) или в виде блок-схемы. В этом случае вы должны использовать те же самые исходные данные и переменные, какие были предложены в условии (например, в образце, записанном на естественном языке).

- 15) Дан целочисленный массив из 30 элементов. Элементы массива могут принимать произвольные целые значения. Опишите на русском языке или на одном из языков программирования алгоритм, который находит и выводит номер третьего положительного элемента массива (если из массива вычеркнуть все неположительные элементы, этот элемент стоял бы в получившемся массиве на третьем месте). Если в массиве меньше, чем три положительных элемента, вывести сообщение об этом. Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать часть из них.

Паскаль	Естественный язык
<pre>const N=30; var a: array [1..N] of integer; i, j, k: integer; begin for i:=1 to N do readln(a[i]); ... end.</pre>	<p>Объявляем массив A из 30 элементов. Объявляем целочисленные переменные i, j, k. В цикле от 1 до 30 вводим элементы массива A с 1-го по 30-й.</p> <p>...</p>
Си	

<pre>#include<stdio.h> int main(void) { const int N=30; int a[N]; int i, k, j; for (i=0; i<N; i++) scanf("%d", &a[i]); ... }</pre>	
---	--

В качестве ответа вам необходимо привести фрагмент программы (или описание алгоритма на естественном языке), который должен находиться на месте многоточия. Вы можете записать решение также на другом языке программирования (укажите название и используемую версию языка программирования, например, *Borland Pascal 7.0*) или в виде блок-схемы. В этом случае вы должны использовать те же самые исходные данные и переменные, какие были предложены в условии (например, в образце, записанном на естественном языке).

- 16) Дан целочисленный массив из 30 элементов. Элементы массива могут принимать произвольные целые значения. Опишите на русском языке или на одном из языков программирования алгоритм, который находит и выводит сумму наибольшей по длине возрастающей последовательности подряд идущих элементов. Если таких последовательностей несколько, можно вывести любую из них. Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать часть из них.

Паскаль	Естественный язык
<pre>const N=30; var a: array [1..N] of integer; i, l, lmax, s, smax: integer; begin for i:=1 to N do readln(a[i]); ... end.</pre>	<p>Объявляем массив A из 30 элементов. Объявляем целочисленные переменные i, l, lmax, s, smax.</p> <p>В цикле от 1 до 30 вводим элементы массива A с 1-го по 30-й.</p> <p>...</p>
Си	
<pre>#include<stdio.h> int main(void) { const int N=30; int a[N]; int i, l, lmax, s, smax; for (i=0; i<N; i++) scanf("%d", &a[i]); ... }</pre>	

В качестве ответа вам необходимо привести фрагмент программы (или описание алгоритма на естественном языке), который должен находиться на месте многоточия. Вы можете записать решение также на другом языке программирования (укажите название и используемую версию языка программирования, например, *Borland Pascal 7.0*) или в виде блок-схемы. В этом случае вы должны использовать те же самые исходные данные и переменные, какие были предложены в условии (например, в образце, записанном на естественном языке).

- 17) Дан целочисленный массив из 30 элементов. Элементы массива могут принимать любые целые значения. Опишите на русском языке или на одном из языков программирования алгоритм, который находит и выводит номер элемента массива, наименее отличающегося от среднего арифметического всех его элементов. Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать часть из них.

Паскаль	Естественный язык
<pre>const N=30; var a: array [1..N] of integer;</pre>	<p>Объявляем массив A из 30 элементов. Объяв-</p>

<pre>i, k: integer; s, min: real; begin for i:=1 to N do readln(a[i]); ... end.</pre>	<p>ляем целочисленные переменные i, k.</p> <p>Объявляем действительные переменные s, min.</p> <p>В цикле от 1 до 30 вводим элементы массива A с 1-го по 30-й.</p> <p>...</p>
Си	
<pre>#include<stdio.h> int main(void) { const int N=30; int a[N]; int i, k; float s, min; for (i=0; i<N; i++) scanf("%d", &a[i]); ... }</pre>	

В качестве ответа вам необходимо привести фрагмент программы (или описание алгоритма на естественном языке), который должен находиться на месте многоточия. Вы можете записать решение также на другом языке программирования (укажите название и используемую версию языка программирования, например, *Borland Pascal 7.0*) или в виде блок-схемы. В этом случае вы должны использовать те же самые исходные данные и переменные, какие были предложены в условии (например, в образце, записанном на естественном языке).

- 18) Дан целочисленный массив из 30 элементов. Элементы массива могут принимать произвольные целые значения. Опишите на русском языке или на одном из языков программирования алгоритм, который находит и выводит номера двух элементов массива, сумма которых минимальна. Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать часть из них.

Паскаль	Естественный язык
<pre>const N=30; var a: array [1..N] of integer; i, j, min, min2, s: integer; begin for i:=1 to N do readln(a[i]); ... end.</pre>	<p>Объявляем массив A из 30 элементов. Объявляем целочисленные переменные i, j, min, min2, s. В цикле от 1 до 30 вводим элементы массива A с 1-го по 30-й.</p> <p>...</p>
Си	
<pre>#include<stdio.h> int main(void) { const int N=30; int a[N]; int i, j, min, min2, s; for (i=0; i<N; i++) scanf("%d", &a[i]); ... }</pre>	

В качестве ответа вам необходимо привести фрагмент программы (или описание алгоритма на естественном языке), который должен находиться на месте многоточия. Вы можете записать решение также на другом языке программирования (укажите название и используемую версию языка программирования, например, *Borland Pascal 7.0*) или в виде блок-схемы. В этом случае вы должны использовать те же самые исходные данные и переменные, какие были предложены в условии (например, в образце, записанном на естественном языке).

- 19) Дан целочисленный массив из 30 элементов. Элементы массива могут принимать произвольные целые значения. Опишите на русском языке или на одном из языков программирования алгоритм, который находит и выводит номера двух элементов массива, наименее отличающихся друг от друга. Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать часть из них.

Паскаль	Естественный язык
<pre>const N=30; var a: array [1..N] of integer; i, j, min, min2, s: integer; begin for i:=1 to N do readln(a[i]); ... end.</pre>	<p>Объявляем массив A из 30 элементов. Объявляем целочисленные переменные i, j, min, min2, s. В цикле от 1 до 30 вводим элементы массива A с 1-го по 30-й.</p> <p>...</p>
Си	
<pre>#include<stdio.h> int main(void) { const int N=30; int a[N]; int i, j, min, min2, s; for (i=0; i<N; i++) scanf("%d", &a[i]); ... }</pre>	

В качестве ответа вам необходимо привести фрагмент программы (или описание алгоритма на естественном языке), который должен находиться на месте многоточия. Вы можете записать решение также на другом языке программирования (укажите название и используемую версию языка программирования, например, *Borland Pascal 7.0*) или в виде блок-схемы. В этом случае вы должны использовать те же самые исходные данные и переменные, какие были предложены в условии (например, в образце, записанном на естественном языке).

- 20) Дан целочисленный массив из 30 элементов. Элементы массива могут принимать значения от -1000 до 1000. Опишите на русском языке или на одном из языков программирования алгоритм, который находит и выводит среднее арифметическое всех элементов массива, оканчивающихся цифрой 5. Гарантируется, что хотя бы один такой элемент в массиве есть. Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать часть из них.

Паскаль	Естественный язык
<pre>const N=30; var a: array [1..N] of integer; i, x, y: integer; s: real; begin for i:=1 to N do readln(a[i]); ... end.</pre>	<p>Объявляем массив A из 30 элементов. Объявляем целочисленные переменные i, x, y и вещественную переменную s. В цикле от 1 до 30 вводим элементы массива A с 1-го по 30-й.</p> <p>...</p>
Си	
<pre>#include<stdio.h> int main(void) { const int N=30; int a[N]; int i, x, y; float s; for (i=0; i<N; i++) scanf("%d", &a[i]); ... }</pre>	

}	
---	--

В качестве ответа вам необходимо привести фрагмент программы (или описание алгоритма на естественном языке), который должен находиться на месте многоточия. Вы можете записать решение также на другом языке программирования (укажите название и используемую версию языка программирования, например, *Borland Pascal 7.0*) или в виде блок-схемы. В этом случае вы должны использовать те же самые исходные данные и переменные, какие были предложены в условии (например, в образце, записанном на естественном языке).

- 21) Дан целочисленный массив из 30 элементов. Элементы массива могут принимать значения от 0 до 1000. Опишите на русском языке или на одном из языков программирования алгоритм, который находит и выводит среднее арифметическое всех элементов массива, имеющих нечётное значение. Гарантируется, что хотя бы один такой элемент в массиве есть.

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

Паскаль	Естественный язык
<pre>const N=30; var a: array [1..N] of integer; i, x, y: integer; s: real; begin for i:=1 to N do readln(a[i]); ... end.</pre>	<p>Объявляем массив A из 30 элементов. Объявляем целочисленные переменные i, x, y и вещественную переменную s. В цикле от 1 до 30 вводим элементы массива A с 1-го по 30-й.</p> <p>...</p>
Си	
<pre>#include<stdio.h> int main(void) { const int N=30; int a[N]; int i, x, y; float s; for (i=0; i<N; i++) scanf("%d", &a[i]); ... }</pre>	

В качестве ответа вам необходимо привести фрагмент программы (или описание алгоритма на естественном языке), который должен находиться на месте многоточия. Вы можете записать решение также на другом языке программирования (укажите название и используемую версию языка программирования, например, *Borland Pascal 7.0*) или в виде блок-схемы. В этом случае вы должны использовать те же самые исходные данные и переменные, какие были предложены в условии (например, в образце, записанном на естественном языке).

- 22) Опишите на русском языке или на одном из языков программирования алгоритм подсчета максимального количества подряд идущих отрицательных элементов в целочисленном массиве длины 30.
- 23) Дан целочисленный массив из 30 элементов. Элементы массива могут принимать целочисленные значения от -20 до 20 — сведения о температуре за каждый день ноября. Опишите на русском языке или на одном из языков программирования алгоритм, который находит и выводит максимальную температуру среди дней, когда были заморозки (температура опускалась ниже нуля). Гарантируется, что хотя бы один день в ноябре была отрицательная температура. Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать часть из них.

Паскаль	Естественный язык
<pre>const N=30; var a: array [1..N] of integer;</pre>	<p>Объявляем массив A из 30 элементов. Объяв-</p>

<pre> i, j, max: integer; begin for i:=1 to N do readln(a[i]); ... end.</pre>	ляем целочисленные переменные i, j, max. В цикле от 1 до 30 вводим элементы массива A с 1-го по 30-й.
Си	
<pre> #include<stdio.h> int main(void) { const int N=30; int a[N]; int i, j, max; for (i=0; i<N; i++) scanf("%d", &a[i]); ... }</pre>	

В качестве ответа вам необходимо привести фрагмент программы (или описание алгоритма на естественном языке), который должен находиться на месте многоточия. Вы можете записать решение также на другом языке программирования (укажите название и используемую версию языка программирования, например, *Borland Pascal 7.0*) или в виде блок-схемы. В этом случае вы должны использовать те же самые исходные данные и переменные, какие были предложены в условии (например, в образце, записанном на естественном языке).

- 24) Опишите на русском языке или одном из языков программирования алгоритм вычисления среднего арифметического элементов квадратной матрицы размером 5 на 5, стоящих на главной диагонали.
- 25) Дан целочисленный массив из 30 элементов, все элементы которого – неотрицательные числа, не превосходящие 10000. Опишите на русском языке или на одном из языков программирования алгоритм, который находит и выводит минимальное трехзначное число, записанное в этом массиве. Если таких чисел нет, нужно вывести сообщение "Таких чисел нет". Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать часть из них.

Паскаль	Естественный язык
<pre> const N=30; var a: array [1..N] of integer; i, j, min: integer; begin for i:=1 to N do readln(a[i]); ... end.</pre>	Объявляем массив A из 30 элементов. Объявляем целочисленные переменные i, j, min. В цикле от 1 до 30 вводим элементы массива A с 1-го по 30-й.
Си	
<pre> #include<stdio.h> int main(void) { const int N=30; int a[N]; int i, j, min; for (i=0; i<N; i++) scanf("%d", &a[i]); ... }</pre>	

В качестве ответа вам необходимо привести фрагмент программы (или описание алгоритма на естественном языке), который должен находиться на месте многоточия. Вы можете записать решение также на другом языке программирования (укажите название и используемую версию языка программирования, например, *Borland Pascal 7.0*) или в виде блок-схемы. В этом случае вы должны использовать те же самые исходные данные и переменные, какие были предложены в условии (например, в образце, записанном на естественном языке).

- 26) Дан целочисленный массив из 30 элементов, все элементы которого – положительные числа, не превосходящие 100. Опишите на русском языке или на одном из языков программирования алгоритм, который находит и выводит номер элемента, имеющего максимальное количество целочисленных делителей (если таких элементов несколько, то номер любого из них). Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать часть из них

Паскаль	Естественный язык
<pre> const N=30; var a: array [1..N] of integer; i, j, k, imax, kmax: integer; begin for i:=1 to N do readln(a[i]); ... end.</pre>	Объявляем массив A из 30 элементов. Объявляем целочисленные переменные i, j, k, imax, kmax. В цикле от 1 до 30 вводим элементы массива A с 1-го по 30-й.
Си	
<pre> #include<stdio.h> int main(void) { const int N=30; int a[N]; int i, j, k, imax, kmax; for(i=0; i<N; i++) scanf("%d", &a[i]); ... }</pre>	

В качестве ответа вам необходимо привести фрагмент программы (или описание алгоритма на естественном языке), который должен находиться на месте многоточия. Вы можете записать решение также на другом языке программирования (укажите название и используемую версию языка программирования, например, *Borland Pascal 7.0*) или в виде блок-схемы. В этом случае вы должны использовать те же самые исходные данные и переменные, какие были предложены в условии (например, в образце, записанном на естественном языке).

- 27) Дан целочисленный массив из 40 элементов, все элементы которого – целые числа в интервале от -500 до 500. Опишите на русском языке или на одном из языков программирования алгоритм, который находит среднее арифметическое всех положительных элементов массива, которые кратны первому элементу (делятся нацело на первый элемент). Гарантируется, что первый элемент массива положительный. Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать часть из них

Паскаль	Естественный язык
<pre> const N=40; var a: array [1..N] of integer; i, x, y: integer; s: real; begin for i:=1 to N do readln(a[i]); ... end.</pre>	Объявляем массив A из 40 элементов. Объявляем целочисленные переменные i, x, y и вещественная переменная s. В цикле от 1 до 40 вводим элементы массива A с 1-го по 40-й.
Си	
<pre> #include<stdio.h> int main(void) { const int N=40; int a[N]; int i, x, y; float s; for(i=0; i<N; i++) scanf("%d", &a[i]); ... }</pre>	

...	
-----	--

В качестве ответа вам необходимо привести фрагмент программы (или описание алгоритма на естественном языке), который должен находиться на месте многоточия. Вы можете записать решение также на другом языке программирования (укажите название и используемую версию языка программирования, например, *Borland Pascal 7.0*) или в виде блок-схемы. В этом случае вы должны использовать те же самые исходные данные и переменные, какие были предложены в условии (например, в образце, записанном на естественном языке).

- 28) Дан целочисленный массив из 20 элементов, все элементы которого – целые числа в интервале от -1000 до 1000. Опишите на русском языке или на одном из языков программирования алгоритм, который находит минимальное значение из всех нечетных элементов массива, которые делятся на 5. Гарантируется, что хотя бы один такой элемент существует. Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать часть из них

Паскаль	Естественный язык
<pre>const N=20; var a: array [1..N] of integer; i,j,min: integer; begin for i:=1 to N do readln(a[i]); ... end.</pre>	<p>Объявляем массив A из 20 элементов. Объявляем целочисленные переменные i, j, min. В цикле от 1 до 20 вводим элементы массива A с 1-го по 20-й.</p> <p>...</p>
Си	
<pre>#include<stdio.h> int main(void) { const int N=20; int a[N]; int i, x, min; for(i=0; i<N; i++) scanf("%d", &a[i]); ... }</pre>	

В качестве ответа вам необходимо привести фрагмент программы (или описание алгоритма на естественном языке), который должен находиться на месте многоточия. Вы можете записать решение также на другом языке программирования (укажите название и используемую версию языка программирования, например, *Borland Pascal 7.0*) или в виде блок-схемы. В этом случае вы должны использовать те же самые исходные данные и переменные, какие были предложены в условии (например, в образце, записанном на естественном языке).

- 29) Дан целочисленный массив из 30 элементов. Элементы массива могут принимать целые значения от 0 до 1000. Опишите на русском языке или на одном из языков программирования алгоритм, позволяющий найти и вывести сумму элементов массива, кратных тринадцати. Гарантируется, что в исходном массиве есть хотя бы один элемент, значение которого делится на тринадцать. Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать часть из них.

Паскаль	Естественный язык
<pre>const N=30; var a: array [1..N] of integer; i,j,s: integer; begin for i:=1 to N do readln(a[i]); ... end.</pre>	<p>Объявляем массив A из 30 элементов. Объявляем целочисленные переменные I, J, S. В цикле от 1 до 30 вводим элементы массива A с 1-го по 30-й.</p> <p>...</p>
Си	

<pre>#include<stdio.h> int main(void) { const int N=30; int a[N]; int i, x, s; for(i=0; i<N; i++) scanf("%d", &a[i]); ... }</pre>	
---	--

В качестве ответа вам необходимо привести фрагмент программы (или описание алгоритма на естественном языке), который должен находиться на месте многоточия. Вы можете записать решение также на другом языке программирования (укажите название и используемую версию языка программирования, например, *FreePascal 2.4*) или в виде блок-схемы. В этом случае вы должны использовать те же самые исходные данные и переменные, какие были предложены в условии (например, в образце, записанном на естественном языке).

- 30) Дан целочисленный массив из 30 элементов. Элементы массива могут принимать целые значения от 0 до 1000. Опишите на русском языке или на одном из языков программирования алгоритм, позволяющий найти среднее арифметическое нечётных трёхзначных чисел, записанных в этом массиве. Если ни одного такого числа нет, нужно вывести сообщение об этом. Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать часть из них.

Паскаль	Естественный язык
<pre>const N=30; var a: array [1..N] of integer; i,j,s: integer; begin for i:=1 to N do readln(a[i]); ... end.</pre>	<p>Объявляем массив A из 30 элементов. Объявляем целочисленные переменные I, J, MIN. В цикле от 1 до 30 вводим элементы массива A с 1-го по 30-й.</p> <p>...</p>
Си	
<pre>#include<stdio.h> int main(void) { const int N=30; int a[N]; int i, x, s; for(i=0; i<N; i++) scanf("%d", &a[i]); ... }</pre>	

В качестве ответа вам необходимо привести фрагмент программы (или описание алгоритма на естественном языке), который должен находиться на месте многоточия. Вы можете записать решение также на другом языке программирования (укажите название и используемую версию языка программирования, например, *FreePascal 2.4*) или в виде блок-схемы. В этом случае вы должны использовать те же самые исходные данные и переменные, какие были предложены в условии (например, в образце, записанном на естественном языке).

- 31) Дан целочисленный массив из 30 элементов. Элементы массива могут принимать целые значения от 0 до 100. Опишите на русском языке или на одном из языков программирования алгоритм, позволяющий найти и вывести произведение элементов массива, которые имеют чётное значение и не оканчиваются на 0. Гарантируется, что в исходном массиве есть хотя бы один элемент, значение которого чётно и не оканчивается на 0. Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать

часть из них. Исходные данные всегда подобраны так, что результат произведения не выходит за пределы объявленных типов данных.

Паскаль	Естественный язык
<pre>const N=30; var a: array [1..N] of longint; i, j, p: longint; begin for i := 1 to N do readln(a[i]); ... end.</pre>	<p>Объявляем массив A из 30 элементов.</p> <p>Объявляем целочисленные переменные I, J, P. В цикле от 1 до 30 вводим элементы массива A с 1-го по 30-й.</p> <p>...</p>
Си	
<pre>#include<stdio.h> int main(void) { const int N=30; int a[N]; int i, j, p; for (i=0; i<N; i++) scanf("%d", &a[i]); ... }</pre>	

В качестве ответа вам необходимо привести фрагмент программы (или описание алгоритма на естественном языке), который должен находиться на месте многоточия. Вы можете записать решение также на другом языке программирования (укажите название и используемую версию языка программирования, например, *FreePascal 2.4*) или в виде блок-схемы. В этом случае вы должны использовать те же самые исходные данные и переменные, какие были предложены в условии (например, в образце, записанном на естественном языке).

- 32) Дан целочисленный массив из 30 элементов. Элементы массива могут принимать целые значения от 0 до 100. Опишите на русском языке или на одном из языков программирования алгоритм, позволяющий найти и вывести произведение двузначных элементов массива, которые не делятся на 6. Гарантируется, что в исходном массиве есть хотя бы один такой элемент. Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать часть из них. Исходные данные всегда подобраны так, что результат произведения не выходит за пределы объявленных типов данных.

Паскаль	Естественный язык
<pre>const N=30; var a: array [1..N] of longint; i, j, p: longint; begin for i := 1 to N do readln(a[i]); ... end.</pre>	<p>Объявляем массив A из 30 элементов.</p> <p>Объявляем целочисленные переменные I, J, P. В цикле от 1 до 30 вводим элементы массива A с 1-го по 30-й.</p> <p>...</p>
Си	
<pre>#include<stdio.h> int main(void) { const int N=30; int a[N]; int i, j, p; for (i=0; i<N; i++) scanf("%d", &a[i]); ... }</pre>	

В качестве ответа вам необходимо привести фрагмент программы (или описание алгоритма на естественном языке), который должен находиться на месте многоточия. Вы можете записать ре-

шение также на другом языке программирования (укажите название и используемую версию языка программирования, например, *FreePascal 2.4*) или в виде блок-схемы. В этом случае вы должны использовать те же самые исходные данные и переменные, какие были предложены в условии (например, в образце, записанном на естественном языке).

- 33) Дан массив, содержащий 70 целых чисел. Опишите на одном из языков программирования алгоритм, позволяющий найти и вывести наименьшее положительное нечетное число, содержащееся в массиве. Гарантируется, что в массиве есть хотя бы одно положительное нечетное число. Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать часть из них.

Паскаль	Естественный язык
<pre>const N=70; var a: array [1..N] of integer; i, j, m: integer; begin for i:=1 to N do readln(a[i]); ... end.</pre>	<p>Объявляем массив A из 70 элементов.</p> <p>Объявляем целочисленные переменные I, J, M. В цикле от 1 до 70 вводим элементы массива A с 1-го по 70-й.</p> <p>...</p>
Си	
<pre>#include<stdio.h> int main(void) { const int N=70; int a[N]; int i, j, m; for (i=0; i<N; i++) scanf("%d", &a[i]); ... }</pre>	

- 34) Дан массив, содержащий 70 целых чисел. Опишите на одном из языков программирования алгоритм, позволяющий найти и вывести наименьшее содержащееся в массиве положительное число, десятичная запись которого оканчивается цифрой 7. Гарантируется, что в массиве есть хотя бы один положительный элемент, десятичная запись которого оканчивается цифрой 7. Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать часть из них.

Паскаль	Естественный язык
<pre>const N=70; var a: array [1..N] of integer; i, j, m: integer; begin for i:=1 to N do readln(a[i]); ... end.</pre>	<p>Объявляем массив A из 70 элементов.</p> <p>Объявляем целочисленные переменные I, J, M. В цикле от 1 до 70 вводим элементы массива A с 1-го по 70-й.</p> <p>...</p>
Си	
<pre>#include<stdio.h> int main(void) { const int N=70; int a[N]; int i, j, m; for (i=0; i<N; i++) scanf("%d", &a[i]); ... }</pre>	

- 35) Дан массив, содержащий 70 целых чисел. Опишите на одном из языков программирования **эффективный** алгоритм, позволяющий найти и вывести наибольшую разность двух чисел, содержащихся в массиве. Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать часть из описанных переменных.

Паскаль	Естественный язык
<pre>const N=70; var a: array [1..N] of integer; i, j, x, y: integer; begin for i:=1 to N do readln(a[i]); ... end.</pre>	<p>Объявляем массив A из 70 элементов.</p> <p>Объявляем целочисленные переменные I, J, X, Y. В цикле от 1 до 70 вводим элементы массива A с 1-го по 70-й.</p> <p>...</p>
Си	
<pre>#include<stdio.h> int main(void) { const int N=70; int a[N]; int i, j, x, y; for (i=0; i<N; i++) scanf("%d", &a[i]); ... }</pre>	

- 36) Дан массив, содержащий 70 неотрицательных целых чисел. Опишите на одном из языков программирования алгоритм, позволяющий найти и вывести наименьшую нечётную сумму двух соседних элементов массива. Гарантируется, что в массиве есть соседние элементы с нечётной суммой. Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать часть из них.

Паскаль	Естественный язык
<pre>const N=70; var a: array [1..N] of integer; i, j, x, y: integer; begin for i:=1 to N do readln(a[i]); ... end.</pre>	<p>Объявляем массив A из 70 элементов.</p> <p>Объявляем целочисленные переменные I, J, X, Y. В цикле от 1 до 70 вводим элементы массива A с 1-го по 70-й.</p> <p>...</p>
Си	
<pre>#include<stdio.h> int main(void) { const int N=70; int a[N]; int i, j, x, y; for (i=0; i<N; i++) scanf("%d", &a[i]); ... }</pre>	

- 37) Дан массив, содержащий 2014 неотрицательных целых чисел, не превышающих 10 000. Опишите на одном из языков программирования алгоритм, позволяющий найти и вывести сумму всех содержащихся в массиве трёхзначных чисел, десятичная запись которых оканчивается на 9, но не на 99. Если подходящих чисел в массиве нет, программа должна вывести число –1. Исходные данные

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

Паскаль	Алгоритмический язык
<pre>const N=2014; var a: array [1..N] of integer; i, j, s: integer; begin for i:=1 to N do readln(a[i]); ... end.</pre>	<pre>алг нач цел N=2014 целтаб a[1:N] цел i, j, s нц для i от 1 до N ввод a[i] кц ... кон</pre>
Си	
<pre>#include<stdio.h> int main(void) { const int N=2014; int a[N]; int i, j, s; for (i=0; i<N; i++) scanf("%d", &a[i]); ... }</pre>	

- 38) Дан массив, содержащий 2014 положительных целых чисел. Напишите на одном из языков программирования программу, которая находит в этом массиве количество элементов, значение которых более чем в два раза превосходит значение предшествующего элемента. Например, для массива из 6 элементов, содержащего числа 2, 5, 10, 15, 40, 100, программа должна вывести ответ 3 (условию соответствуют элементы со значениями 5, 40 и 100). Программа должна вывести общее количество подходящих элементов, значения элементов выводить не нужно. Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать часть из описанных переменных.

Паскаль	Алгоритмический язык
<pre>const N = 2014; var a: array [1..N] of integer; i, j, k: integer; begin for i:=1 to N do readln(a[i]); ... end.</pre>	<pre>алг нач цел N=2014 целтаб a[1:N] цел i, j, k нц для i от 1 до N ввод a[i] кц ... кон</pre>
Си	
<pre>#include<stdio.h> int main(void) { const int N=2014; int a[N]; int i, k, j; for (i=0; i<N; i++) scanf("%d", &a[i]); ... }</pre>	

39) Дан массив, содержащий 2014 положительных целых чисел. Симметричной парой называются два элемента, которые находятся на равном расстоянии от концов массива. Например, 1-й и 2014-й элементы, 2-й и 2013-й и т. д. Порядок элементов в симметричной паре не учитывается: элементы на 1 и 2014 местах – это та же самая пара, что и элементы на 2014 и 1 местах. Напишите на одном из языков программирования программу, которая подсчитывает в массиве количество симметричных пар, у которых сумма элементов больше 20. Программа должна вывести одно число – количество отобранных симметричных пар. Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать часть из описанных переменных.

Паскаль	Алгоритмический язык
<pre>const N=2014; var a: array [1..N] of integer; i, j, k: integer; begin for i:=1 to N do readln(a[i]); ... end.</pre>	<pre>алг нач цел N=2014 целтаб a[1:N] цел i, j, k нц для i от 1 до N ввод a[i] кц ... кон</pre>
Си	
<pre>#include<stdio.h> int main(void) { const int N= 2014; int a[N]; int i, k, j; for (i=0; i<N; i++) scanf("%d", &a[i]); ... }</pre>	

40) Дан массив, содержащий 2014 положительных целых чисел. Напишите на одном из языков программирования программу, которая находит в этом массиве количество локальных минимумов. Локальным минимумом называется элемент массива, который меньше всех своих соседей. Например, в массиве из 6 элементов, содержащем числа 4, 6, 12, 7, 3, 8, есть два локальных минимума: это элементы, равные 4 и 3. Программа должна вывести общее количество подходящих элементов, значения элементов выводить не нужно. Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать часть из описанных.

Паскаль	Алгоритмический язык
<pre>const N=2014; var a: array [1..N] of integer; i, j, k: integer; begin for i:=1 to N do readln(a[i]); ... end.</pre>	<pre>алг нач цел N=2014 целтаб a[1:N] цел i, j, k нц для i от 1 до N ввод a[i] кц ... кон</pre>
Си	
<pre>#include<stdio.h></pre>	

<pre>int main(void) { const int N= 2014; int a[N]; int i, k, j; for (i=0; i<N; i++) scanf("%d", &a[i]); ... }</pre>	
---	--

41) Дан массив, содержащий 2014 целых чисел в диапазоне от -10000 до 10000. Напишите на одном из языков программирования программу, которая находит в этом массиве количество пар соседних элементов массива, произведение которых нечётно, а сумма – положительна. Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать часть из описанных.

Паскаль	Алгоритмический язык
<pre>const N=2014; var a: array [1..N] of integer; i, j, k: longint; begin for i:=1 to N do readln(a[i]); ... end.</pre>	<pre>алг нач цел N=2014 целтаб a[1:N] цел i, j, k нц для i от 1 до N ввод a[i] кц ... кон</pre>
Си	
<pre>#include<stdio.h> int main(void) { const int N=2014; int a[N]; int i, j, k; for (i=0; i<N; i++) scanf("%d", &a[i]); ... }</pre>	

42) Дан целочисленный массив из 20 элементов. Элементы массива могут принимать целые значения от -10 000 до 10 000 включительно. Опишите на одном из языков программирования алгоритм, позволяющий найти и вывести количество пар элементов массива, в которых сумма элементов делится на 2, но не делится на 4. В данной задаче под парой подразумеваются два соседних элемента массива.

Паскаль	Алгоритмический язык
<pre>const N=20; var a: array [1..N] of integer; i, j, k: longint; begin for i:=1 to N do readln(a[i]); ... end.</pre>	<pre>алг нач цел N=20 целтаб a[1:N] цел i, j, k нц для i от 1 до N ввод a[i] кц ... кон</pre>
Си	

```
#include<stdio.h>
int main(void) {
const int N=20;
int a[N];
int i, j, k;
for (i=0; i<N; i++)
scanf("%d", &a[i]);
...
}
```

- 43) Дан целочисленный массив из 40 элементов. Элементы массива могут принимать целые значения от –100 до 100 включительно. Опишите на естественном языке или на одном из языков программирования алгоритм, позволяющий найти и вывести количество пар элементов массива, произведение которых положительно, а сумма кратна 7. Под парой подразумевается два подряд идущих элемента массива.

Паскаль	Алгоритмический язык
<pre>const N=40; var a: array [1..N] of integer; i, j, k: longint; begin for i:=1 to N do readln(a[i]); ... end.</pre>	<pre>алг нач цел N=40 целтаб a[1:N] цел i, j, k нц для i от 1 до N ввод a[i] кц ... кон</pre>
Си	
<pre>#include<stdio.h> int main(void) { const int N=40; int a[N]; int i, j, k; for (i=0; i<N; i++) scanf("%d", &a[i]); ... }</pre>	

- 44) Дан целочисленный массив из 40 элементов. Элементы массива могут принимать целые значения от 0 до 100 включительно. Опишите на естественном языке или на одном из языков программирования алгоритм, позволяющий найти и вывести количество пар элементов массива, сумма которых не кратна 6, а произведение меньше 1000. Под парой подразумевается два подряд идущих элемента массива.

Паскаль	Алгоритмический язык
<pre>const N=40; var a: array [1..N] of integer; i, j, k: longint; begin for i:=1 to N do readln(a[i]); ... end.</pre>	<pre>алг нач цел N=40 целтаб a[1:N] цел i, j, k нц для i от 1 до N ввод a[i] кц ... кон</pre>
Си	

```
#include<stdio.h>
int main(void) {
const int N= 40;
int a[N];
int i, j, k;
for (i=0; i<N; i++)
scanf("%d", &a[i]);
...
}
```

- 45) Дан массив, содержащий неотрицательные целые числа. Необходимо вывести:
– максимальный чётный элемент, если количество чётных элементов не меньше, чем нечётных;
– максимальный нечётный элемент, если количество нечётных элементов больше, чем чётных.
Например, для массива из шести элементов, равных соответственно 4, 6, 12, 17, 3, 8, ответом будет 12 – наибольшее чётное число, поскольку чётных чисел в этом массиве больше.

Паскаль	Алгоритмический язык
<pre>const N=2000; var a: array [1..N] of integer; i, j, k, m: longint; begin for i:=1 to N do readln(a[i]); ... end.</pre>	<pre>алг нач цел N=2000 целтаб a[1:N] цел i, j, k, m нц для i от 1 до N ввод a[i] кц ... кон</pre>
Си	
<pre>#include<stdio.h> int main(void) { const int N=2000; int a[N]; int i, j, k, m; for (i=0; i<N; i++) scanf("%d", &a[i]); ... }</pre>	

- 46) Дан массив, содержащий неотрицательные целые числа, не превышающие 10 000. Необходимо вывести:
– минимальный чётный элемент, если количество чётных элементов не больше, чем нечётных;
– минимальный нечётный элемент, если количество нечётных элементов меньше, чем чётных.
Например, для массива из шести элементов, равных соответственно 4, 6, 12, 17, 9, 8, ответом будет 9 – наименьшее нечётное число, поскольку нечётных чисел в этом массиве меньше.

Паскаль	Алгоритмический язык
<pre>const N=2000; var a: array [1..N] of integer; i, j, k, m: longint; begin for i:=1 to N do readln(a[i]); ... end.</pre>	<pre>алг нач цел N=2000 целтаб a[1:N] цел i, j, k, m нц для i от 1 до N ввод a[i] кц ... кон</pre>
Си	

<pre>#include<stdio.h> int main(void) { const int N=2000; int a[N]; int i, j, k, m; for (i=0; i<N; i++) scanf("%d", &a[i]); ... }</pre>	
--	--

- 47) Дан целочисленный массив из 20 элементов. Элементы массива могут принимать целые значения от –10 000 до 10 000 включительно. Опишите на естественном языке или на одном из языков программирования алгоритм, позволяющий найти и вывести количество пар элементов массива, в которых оба числа делятся на 3. В данной задаче под парой подразумевается два подряд идущих элемента массива. Например, для массива из пяти элементов: 6; 2; –3; 6 – ответ: 2.

Паскаль	Алгоритмический язык
<pre>const N = 20; var a: array [1..N] of integer; i, j, k: integer; begin for i := 1 to N do readln(a[i]); ... end.</pre>	<pre>алг нач цел N = 20 целтаб a[1:N] цел i, j, k нц для i от 1 до N ввод a[i] кц ... кон</pre>
Си	
<pre>#include <stdio.h> #define N 20 int main() { int a[N]; int i, j, k; for (i = 0; i<N; i++) scanf("%d", &a[i]); ... return 0; }</pre>	

- 48) Дан целочисленный массив из 2000 элементов. Если сумма всех элементов массива чётная, нужно вывести количество нечётных (по значению) элементов массива, если нечётная – количество чётных. Например, для массива из 6 элементов, равных соответственно 2, 6, 12, 17, 3, 8, ответом будет 2 – количество нечётных элементов, так как общая сумма всех элементов чётна.

Паскаль	Алгоритмический язык
<pre>const N = 2000; var a: array [1..N] of integer; i, k: integer; begin for i := 1 to N do readln(a[i]); ... end.</pre>	<pre>алг нач цел N = 2000 целтаб a[1:N] цел i, k нц для i от 1 до N ввод a[i] кц ... кон</pre>
Си	

<pre>#include <stdio.h> #define N 2000 int main() { int a[N]; int i, k; for (i = 0; i<N; i++) scanf("%d", &a[i]); ... return 0; }</pre>	
--	--

- 49) Дан целочисленный массив из 20 элементов. Элементы массива могут принимать целые значения от –10 000 до 10 000 включительно. Опишите на естественном языке или на одном из языков программирования алгоритм, позволяющий найти и вывести количество пар элементов массива, в которых хотя бы одно число делится на 3. В данной задаче под парой подразумевается два подряд идущих элемента массива.

Паскаль	Алгоритмический язык
<pre>const N = 20; var a: array [1..N] of integer; i, j, k: integer; begin for i := 1 to N do readln(a[i]); ... end.</pre>	<pre>алг нач цел N = 20 целтаб a[1:N] цел i, j, k нц для i от 1 до N ввод a[i] кц ... кон</pre>
Си	
<pre>#include <stdio.h> #define N 20 int main() { int a[N]; int i, j, k; for (i = 0; i<N; i++) scanf("%d", &a[i]); ... return 0; }</pre>	

- 50) Дан целочисленный массив из 40 элементов. Элементы массива могут принимать целые значения от 0 до 10000 включительно. Опишите на естественном языке или на одном из языков программирования алгоритм, позволяющий найти и вывести количество пар элементов массива, в которых десятичная запись хотя бы одного числа оканчивается на 5.

Паскаль	Алгоритмический язык
<pre>const N = 40; var a: array [1..N] of integer; i, j, k: integer; begin for i := 1 to N do readln(a[i]); ... end.</pre>	<pre>алг нач цел N = 40 целтаб a[1:N] цел i, j, k нц для i от 1 до N ввод a[i] кц ... кон</pre>
Си	Python

<pre>#include <stdio.h> #define N 40 int main() { int a[N]; int i, j, k; for (i = 0; i < N; i++) scanf("%d", &a[i]); ... return 0; }</pre>	<p>#допускается также использовать #две целочисленные переменные #j и k</p> <pre>a = [] n = 40 for i in range(n): a.append(int(input()))</pre>
---	--

- 51) Дан целочисленный массив из 40 элементов. Элементы массива могут принимать целые значения от 1 до 10000 включительно. Опишите на естественном языке или на одном из языков программирования алгоритм, позволяющий найти и вывести наибольший из элементов массива, шестнадцатеричная запись которого оканчивается на букву F. Если таких чисел нет, нужно вывести ответ 0.

Паскаль	Алгоритмический язык
<pre>const N = 40; var a: array [1..N] of integer; i, j, k: integer; begin for i := 1 to N do readln(a[i]); ... end.</pre>	<pre>алг нач цел N = 40 целтаб a[1:N] цел i, j, k нц для i от 1 до N ввод a[i] кц ... кон</pre>
Си	Python
<pre>#include <stdio.h> #define N 40 int main() { int a[N]; int i, j, k; for (i = 0; i < N; i++) scanf("%d", &a[i]); ... return 0; }</pre>	<p>#допускается также использовать #две целочисленные переменные #j и k</p> <pre>a = [] n = 40 for i in range(n): a.append(int(input()))</pre>

- 52) Дан целочисленный массив из 40 элементов. Элементы массива могут принимать целые значения от 1 до 10000 включительно. Опишите на естественном языке или на одном из языков программирования алгоритм, позволяющий найти и вывести наименьший из элементов массива, восьмеричная запись которого оканчивается на цифру 7. Если таких чисел нет, нужно вывести ответ 0.

Паскаль	Алгоритмический язык
<pre>const N = 40; var a: array [1..N] of integer; i, j, k: integer; begin for i := 1 to N do readln(a[i]); ... end.</pre>	<pre>алг нач цел N = 40 целтаб a[1:N] цел i, j, k нц для i от 1 до N ввод a[i] кц ... кон</pre>
Си	Python

<pre>#include <stdio.h> #define N 40 int main() { int a[N]; int i, j, k; for (i = 0; i < N; i++) scanf("%d", &a[i]); ... return 0; }</pre>	<p>#допускается также использовать #две целочисленные переменные #j и k</p> <pre>a = [] n = 40 for i in range(n): a.append(int(input()))</pre>
---	--

- 53) Дан целочисленный массив из 40 элементов. Элементы массива могут принимать целые значения от 1 до 10000 включительно. Опишите на естественном языке или на одном из языков программирования алгоритм, позволяющий найти и вывести наибольший из элементов массива, восьмеричная запись которого содержит ровно три цифры. Если таких чисел нет, нужно вывести ответ 0.

Паскаль	Алгоритмический язык
<pre>const N = 40; var a: array [1..N] of integer; i, j, k: integer; begin for i := 1 to N do readln(a[i]); ... end.</pre>	<pre>алг нач цел N = 40 целтаб a[1:N] цел i, j, k нц для i от 1 до N ввод a[i] кц ... кон</pre>
Си	Python
<pre>#include <stdio.h> #define N 40 int main() { int a[N]; int i, j, k; for (i = 0; i < N; i++) scanf("%d", &a[i]); ... return 0; }</pre>	<p>#допускается также использовать #две целочисленные переменные #j и k</p> <pre>a = [] n = 40 for i in range(n): a.append(int(input()))</pre>

- 54) Дан целочисленный массив из 40 элементов. Элементы массива могут принимать целые значения от 1 до 10000 включительно. Опишите на естественном языке или на одном из языков программирования алгоритм, позволяющий найти и вывести наименьший из элементов массива, шестнадцатеричная запись которого содержит ровно две цифры. Если таких чисел нет, нужно вывести ответ 0.

Паскаль	Алгоритмический язык
<pre>const N = 40; var a: array [1..N] of integer; i, j, k: integer; begin for i := 1 to N do readln(a[i]); ... end.</pre>	<pre>алг нач цел N = 40 целтаб a[1:N] цел i, j, k нц для i от 1 до N ввод a[i] кц ... кон</pre>
Си	Python

<pre>#include <stdio.h> #define N 40 int main() { int a[N]; int i, j, k; for (i = 0; i < N; i++) scanf("%d", &a[i]); ... return 0; }</pre>	<p>#допускается также использовать #две целочисленные переменные #j и k</p> <pre>a = [] n = 40 for i in range(n): a.append(int(input()))</pre>
---	--

- 55) Дан целочисленный массив из 40 элементов. Элементы массива могут принимать целые значения от 1 до 10000 включительно. Опишите на естественном языке или на одном из языков программирования алгоритм, позволяющий найти и вывести наибольший из элементов массива, восьмеричная запись которого содержит не менее трёх цифр и оканчивается на 5. Если таких чисел нет, нужно вывести ответ 0.

Паскаль	Алгоритмический язык
<pre>const N = 40; var a: array [1..N] of integer; i, j, k: integer; begin for i := 1 to N do readln(a[i]); ... end.</pre>	<pre>алг нач цел N = 40 целтаб a[1:N] цел i, j, k нц для i от 1 до N ввод a[i] кц ... кон</pre>
Си	Python
<pre>#include <stdio.h> #define N 40 int main() { int a[N]; int i, j, k; for (i = 0; i < N; i++) scanf("%d", &a[i]); ... return 0; }</pre>	<p>#допускается также использовать #две целочисленные переменные #j и k</p> <pre>a = [] n = 40 for i in range(n): a.append(int(input()))</pre>

- 56) Дан целочисленный массив из 40 элементов. Элементы массива могут принимать целые значения от 1 до 10000 включительно. Опишите на естественном языке или на одном из языков программирования алгоритм, позволяющий найти и вывести наименьший из элементов массива, шестнадцатеричная запись которого содержит не менее трёх цифр и оканчивается на букву С. Если таких чисел нет, нужно вывести ответ 0.

Паскаль	Алгоритмический язык
<pre>const N = 40; var a: array [1..N] of integer; i, j, k: integer; begin for i := 1 to N do readln(a[i]); ... end.</pre>	<pre>алг нач цел N = 40 целтаб a[1:N] цел i, j, k нц для i от 1 до N ввод a[i] кц ... кон</pre>

Си	Python
<pre>#include <stdio.h> #define N 40 int main() { int a[N]; int i, j, k; for (i = 0; i < N; i++) scanf("%d", &a[i]); ... return 0; }</pre>	<p>#допускается также использовать #две целочисленные переменные #j и k</p> <pre>a = [] n = 40 for i in range(n): a.append(int(input()))</pre>

- 57) Дан целочисленный массив из 40 элементов. Элементы массива могут принимать целые значения от 1 до 10000 включительно. Опишите на естественном языке или на одном из языков программирования алгоритм, позволяющий найти и вывести количество элементов массива, восьмеричная запись которого содержит ровно две цифры, причём первая (старшая) цифра меньше второй (младшей).

Паскаль	Алгоритмический язык
<pre>const N = 40; var a: array [1..N] of integer; i, j, k: integer; begin for i := 1 to N do readln(a[i]); ... end.</pre>	<pre>алг нач цел N = 40 целтаб a[1:N] цел i, j, k нц для i от 1 до N ввод a[i] кц ... кон</pre>
Си	Python
<pre>#include <stdio.h> #define N 40 int main() { int a[N]; int i, j, k; for (i = 0; i < N; i++) scanf("%d", &a[i]); ... return 0; }</pre>	<p>#допускается также использовать #две целочисленные переменные #j и k</p> <pre>a = [] n = 40 for i in range(n): a.append(int(input()))</pre>

- 58) Дан целочисленный массив из 40 элементов. Элементы массива могут принимать целые значения от 1 до 10000 включительно. Опишите на естественном языке или на одном из языков программирования алгоритм, позволяющий найти и вывести минимальный элемент массива, шестнадцатеричная запись которого содержит ровно две цифры, причём первая (старшая) цифра больше второй (младшей). Если таких чисел нет, нужно вывести ответ 0.

Паскаль	Алгоритмический язык
<pre>const N = 40; var a: array [1..N] of integer; i, j, k: integer; begin for i := 1 to N do readln(a[i]); ... end.</pre>	<pre>алг нач цел N = 40 целтаб a[1:N] цел i, j, k нц для i от 1 до N ввод a[i] кц ... кон</pre>

Си	Python
<pre>#include <stdio.h> #define N 40 int main() { int a[N]; int i, j, k; for (i = 0; i < N; i++) scanf("%d", &a[i]); ... return 0; }</pre>	<pre>#допускается также использовать #две целочисленные переменные #j и k a = [] n = 40 for i in range(n): a.append(int(input()))</pre>

- 59) (Д.В. Богданов) Дан массив, содержащий 2017 целых чисел. Необходимо определить количество «уникальных» значений элементов этого массива, то есть количество значений элементов без учёта их возможных повторов. Например, в массиве из 6 элементов, равных соответственно 2; 0; 1; 0; 0; 1, уникальными значениями являются 0; 1; 2, значит, ответом будет 3 — их количество. Исходные данные объявлены так, как показано ниже. Запрещается использовать не описанные переменные, но допускается не использовать некоторые из них.

Паскаль	Алгоритмический язык
<pre>const N = 2017; var a : array [0..N - 1] of integer; i, j, count : integer; begin for i := 0 to N - 1 do readln(a[i]); ... end.</pre>	<pre>алг нач цел N = 2017 целтаб a[0:N-1] цел i, j, count нц для i от 0 до N-1 ввод a[i] кц ... кон</pre>
Си	Python
<pre>#include <stdio.h> #define N 2017 int main() { int a[N]; int i, j, count; for (i = 0; i < N; i++) scanf("%d", &a[i]); ... return 0; }</pre>	<pre># допускается также использо- # вание целочисленных # переменных j, count N = 2017 a = [] for i in range(N): a.append(int(input())) ...</pre>

- 60) Дан целочисленный массив из 40 элементов. Элементы массива могут принимать целые значения от 1 до 10000 включительно. Опишите на естественном языке или на одном из языков программирования алгоритм, позволяющий найти и вывести минимальный элемент массива, шестнадцатеричная запись которого содержит ровно две цифры, причём вторая (младшая) цифра — это буква (от A до F). Если таких чисел нет, нужно вывести ответ 0.

Паскаль	Алгоритмический язык
<pre>const N = 40; var a : array [1..N] of integer; i, j, k : integer; begin for i := 1 to N do readln(a[i]); ... end.</pre>	<pre>алг нач цел N = 40 целтаб a[1:N] цел i, j, k нц для i от 1 до N ввод a[i] кц ... кон</pre>

C++	Python
<pre>#include <iostream> using namespace std; const int N = 40; int main() { int a[N]; int i, k, s; for (i = 0; i < N; i++) cin >> a[i]; ... return 0; }</pre>	<pre>#допускается также использовать #две целочисленные переменные #j и k a = [] n = 40 for i in range(n): a.append(int(input()))</pre>

- 61) Дан массив, содержащий 2018 положительных целых чисел, не превышающих 30 000. Необходимо найти в этом массиве количество элементов, которые кратны 3, а их десятичная запись заканчивается цифрой 1, и заменить каждый из таких элементов на это количество. Напишите программу для решения этой задачи. В качестве результата программа должна вывести изменённый массив, по одному элементу в строке. Например, для исходного массива из 5 элементов
- 15 71 21 111 41
- программа должна вывести (по одному числу в строке) числа
- 15 71 2 2 41.

Паскаль	Алгоритмический язык
<pre>const N = 40; var a : array [1..N] of integer; i, k, s : integer; begin for i := 1 to N do readln(a[i]); ... end.</pre>	<pre>алг нач цел N = 40 целтаб a[1:N] цел i, k, s нц для i от 1 до N ввод a[i] кц ... кон</pre>
C++	Python
<pre>#include <iostream> using namespace std; const int N = 40; int main() { int a[N]; int i, k, s; for (i = 0; i < N; i++) cin >> a[i]; ... return 0; }</pre>	<pre>#допускается также использовать #две целочисленные переменные #k и s a = [] n = 40 for i in range(n): a.append(int(input()))</pre>

- 62) Дан массив, содержащий 40 положительных целых чисел, не превышающих 1000. Необходимо уменьшить все элементы, которые больше 50, на одно и то же значение, при этом минимальный из них должен стать равным 50. Напишите на одном из языков программирования программу для решения этой задачи. В качестве результата программа должна вывести изменённый массив, по одному элементу в строке. Например, для исходного массива из 5 элементов
- 15 71 21 111 41
- программа должна вывести (по одному числу в строке) числа

15 50 21 90 41

(все элементы, большие 50, уменьшены на 21, минимальный из них стал равен 50).

Паскаль	Алгоритмический язык
<pre>const N = 40; var a: array [1..N] of integer; i, k, s: integer; begin for i := 1 to N do readln(a[i]); ... end.</pre>	<pre>алг нач цел N = 40 целтаб a[1:N] цел i, k, s нц для i от 1 до N ввод a[i] кц ... кон</pre>
C++	Python
<pre>#include <iostream> using namespace std; const int N = 40; int main() { int a[N]; int i, k, s; for (i = 0; i < N; i++) cin >> a[i]; ... return 0; }</pre>	<pre>#допускается также использовать #две целочисленные переменные #k и s a = [] n = 40 for i in range(n): a.append(int(input()))</pre>

- 63) Дан массив, содержащий 40 положительных целых чисел, не превышающих 1000. Необходимо уменьшить все элементы, которые больше 80, на одно и то же значение, при этом максимальный из них должен стать равным 80. Напишите на одном из языков программирования программу для решения этой задачи. В качестве результата программа должна вывести изменённый массив, по одному элементу в строке. Например, для исходного массива из 5 элементов

15 91 21 111 41

программа должна вывести (по одному числу в строке) числа

15 60 21 80 41

(все элементы, большие 80, уменьшены на 31, максимальный из них стал равен 80).

Паскаль	Алгоритмический язык
<pre>const N = 40; var a: array [1..N] of integer; i, k, s: integer; begin for i := 1 to N do readln(a[i]); ... end.</pre>	<pre>алг нач цел N = 40 целтаб a[1:N] цел i, k, s нц для i от 1 до N ввод a[i] кц ... кон</pre>
C++	Python
<pre>#include <iostream> using namespace std; const int N = 40; int main() { int a[N]; int i, k, s; for (i = 0; i < N; i++)</pre>	<pre>#допускается также использовать #две целочисленные переменные #k и s a = [] n = 40 for i in range(n):</pre>

<pre>cin >> a[i]; ... return 0; }</pre>	<pre>a.append(int(input()))</pre>
---	-----------------------------------

- 64) Дан массив, содержащий 2018 положительных целых чисел, не превышающих 30000. Необходимо уменьшить (или увеличить) все чётные элементы на одно и то же значение, так чтобы при этом минимальный из них стал равен 20. Напишите на одном из языков программирования программу для решения этой задачи. В качестве результата программа должна вывести изменённый массив, по одному элементу в строке. Например, для исходного массива из 5 элементов

16 91 22 112 41

программа должна вывести (по одному числу в строке) числа

20 91 26 116 41

(все чётные элементы увеличены на 4, минимальный из них стал равен 20).

Паскаль	Алгоритмический язык
<pre>const N = 2018; var a: array [1..N] of integer; i, k, s: integer; begin for i := 1 to N do readln(a[i]); ... end.</pre>	<pre>алг нач цел N = 2018 целтаб a[1:N] цел i, k, s нц для i от 1 до N ввод a[i] кц ... кон</pre>
C++	Python
<pre>#include <iostream> using namespace std; const int N = 2018; int main() { int a[N]; int i, k, s; for (i = 0; i < N; i++) cin >> a[i]; ... return 0; }</pre>	<pre>#допускается также использовать #две целочисленные переменные #k и s a = [] n = 2018 for i in range(n): a.append(int(input()))</pre>

- 65) Дан массив, содержащий 2018 положительных целых чисел, не превышающих 30000. Необходимо уменьшить (или увеличить) все нечётные элементы на одно и то же значение, так чтобы при этом максимальный из них стал равен 1. Напишите на одном из языков программирования программу для решения этой задачи. В качестве результата программа должна вывести изменённый массив, по одному элементу в строке. Например, для исходного массива из 5 элементов

16 91 22 112 41

программа должна вывести (по одному числу в строке) числа

16 1 22 112 -49

(все нечётные элементы уменьшены на 90, максимальный из них стал равен 1).

Паскаль	Алгоритмический язык
<pre>const N = 2018; var a: array [1..N] of integer; i, k, s: integer; begin for i := 1 to N do readln(a[i]);</pre>	<pre>алг нач цел N = 2018 целтаб a[1:N] цел i, k, s нц для i от 1 до N</pre>

... end.	ввод a[i] кц ... кон
C++	Python
<pre>#include <iostream> using namespace std; const int N = 2018; int main() { int a[N]; int i, k, s; for (i = 0; i < N; i++) cin >> a[i]; ... return 0; }</pre>	<pre>#допускается также использовать #две целочисленные переменные #k и s a = [] n = 2018 for i in range(n): a.append(int(input()))</pre>

- 66) (А.Г. Минак) Дан массив, содержащий 2018 положительных целых чисел, не превышающих 10000. Необходимо умножить все элементы массива, меньшие 40, на одно и то же значение, так чтобы при этом максимальный из них стал как можно больше, но не превысил значения 10000. Напишите на одном из языков программирования программу для решения этой задачи. В качестве результата программа должна вывести изменённый массив, по одному элементу в строке. Например, для исходного массива из 5 элементов

1 21 315 81 57

программа должна вывести (по одному числу в строке) числа

476 9996 315 81 57

(все элементы, меньшие 40 умножены на 476).

Паскаль	Алгоритмический язык
<pre>const N = 2018; var a: array [1..N] of integer; i, k, s: integer; begin for i := 1 to N do readln(a[i]); ... end.</pre>	<pre>алг нач цел N = 2018 целтаб a[1:N] цел i, k, s нц для i от 1 до N ввод a[i] кц ... кон</pre>
C++	Python
<pre>#include <iostream> using namespace std; const int N = 2018; int main() { int a[N]; int i, k, s; for (i = 0; i < N; i++) cin >> a[i]; ... return 0; }</pre>	<pre>#допускается также использовать #две целочисленные переменные #k и s a = [] n = 2018 for i in range(n): a.append(int(input()))</pre>

- 67) (Досрочный экзамен 2018 г.) Дан целочисленный массив из 30 элементов. Элементы массива могут принимать целые значения от 0 до 10 000 включительно. Опишите на одном из языков программирования алгоритм, который находит сумму элементов массива, меньших 200 и при этом кратных 5, а затем заменяет каждый такой элемент на число, равное найденной сумме. Гаранти-

руется, что хотя бы один такой элемент в массиве есть. В качестве результата необходимо вывести изменённый массив, каждый элемент выводится с новой строчки. Например, для исходного массива из шести элементов:

204 115 27 20 305 4

программа должна вывести (по одному числу в строке) числа

204 135 27 135 305 4

Паскаль	Алгоритмический язык
<pre>const N = 30; var a: array [1..N] of integer; i, k, s: integer; begin for i := 1 to N do readln(a[i]); ... end.</pre>	<pre>алг нач цел N = 30 целтаб a[1:N] цел i, k, s нц для i от 1 до N ввод a[i] кц ... кон</pre>
C++	Python
<pre>#include <iostream> using namespace std; const int N = 30; int main() { int a[N]; int i, k, s; for (i = 0; i < N; i++) cin >> a[i]; ... return 0; }</pre>	<pre>#допускается также использовать #две целочисленные переменные #k и s a = [] n = 30 for i in range(n): a.append(int(input()))</pre>

- 68) Дан целочисленный массив из 2018 элементов, в котором записаны значения последовательных результатов измерений. Элементы массива могут принимать целые значения от 0 до 10 000 включительно. Опишите на одном из языков программирования алгоритм, который выполняет сглаживание последовательности: каждый элемент заменяется на целочисленное среднее арифметическое данного элемента и его ближайших соседей. Считается, что у крайних элементов только один сосед, а у остальных элементов – по два. Например, для исходного массива из шести элементов:

1 9 2 10 3 11

программа должна вывести (по одному числу в строке) числа

5 4 7 5 8 7

Паскаль	Алгоритмический язык
<pre>const N = 2018; var a: array [1..N] of integer; i, j, k: integer; begin for i := 1 to N do readln(a[i]); ... end.</pre>	<pre>алг нач цел N = 2018 целтаб a[1:N] цел i, j, k нц для i от 1 до N ввод a[i] кц ... кон</pre>
C++	Python
<pre>#include <iostream> using namespace std;</pre>	<pre>#допускается также использовать #две целочисленные переменные</pre>

<pre>const int N = 2018; int main() { int a[N]; int i, j, k; for (i = 0; i < N; i++) cin >> a[i]; ... return 0; }</pre>	<pre>#k и j a = [] n = 2018 for i in range(n): a.append(int(input()))</pre>
--	---

- 69) Дан целочисленный массив из 200 элементов, в котором записаны значения последовательных результатов измерений. Элементы массива могут принимать целые значения от 0 до 10 000 включительно. Опишите на одном из языков программирования алгоритм, который выполняет сглаживание массива: для каждой четвёрки элементов определяет целочисленное среднее арифметическое, и заменяет эти 4 элемента на их среднее. Например, для исходного массива из восьми элементов:

6 0 6 10 11 7 5 9

программа должна вывести (по одному числу в строке) числа

5 5 5 5 8 8 8

Паскаль	Алгоритмический язык
<pre>const N = 200; var a: array [1..N] of integer; i, j, k: integer; begin for i := 1 to N do readln(a[i]); ... end.</pre>	<pre>алг нач цел N = 200 целтаб a[1:N] цел i, j, k нц для i от 1 до N ввод a[i] кц ... кон</pre>
C++	Python
<pre>#include <iostream> using namespace std; const int N = 200; int main() { int a[N]; int i, j, k; for (i = 0; i < N; i++) cin >> a[i]; ... return 0; }</pre>	<pre>#допускается также использовать #две целочисленные переменные #k и j a = [] n = 200 for i in range(n): a.append(int(input()))</pre>

- 70) Дан целочисленный массив из 200 элементов, в котором записаны значения последовательных результатов измерений. Элементы массива могут принимать целые значения от 0 до 10 000 включительно. Опишите на одном из языков программирования алгоритм, который выполняет выборку данных в массиве: заменяет на 0 значения всех элементов, которые отличаются от среднего арифметического всех четных элементов более, чем на половину этого значения. Гарантируется, что хотя бы один чётный положительный элемент в массиве есть. Например, для исходного массива из восьми элементов:

6 1 6 10 11 7 2 9

программа должна вывести (по одному числу в строке) числа

6 0 6 0 0 7 0 9

Паскаль	Алгоритмический язык
---------	----------------------

<pre>const N = 200; var a: array [1..N] of integer; i, k: integer; s: real; begin for i := 1 to N do readln(a[i]); ... end.</pre>	<pre>алг нач цел N = 200 целтаб a[1:N] цел i, k вещ s нц для i от 1 до N ввод a[i] кц ... кон</pre>
C++	Python
<pre>#include <iostream> using namespace std; const int N = 200; int main() { int a[N]; int i, k; float s; for (i = 0; i < N; i++) cin >> a[i]; ... return 0; }</pre>	<pre>#допускается также использовать #две целочисленные переменные #k и s a = [] n = 200 for i in range(n): a.append(int(input()))</pre>

- 71) Дан целочисленный массив из 200 элементов, в котором записаны значения последовательных результатов измерений. Элементы массива могут принимать целые значения от 0 до 10 000 включительно. Опишите на одном из языков программирования алгоритм, который выполняет выборку данных в массиве: если максимальный чётный элемент массива меньше максимального нечётного, все чётные элементы заменяются на нули, иначе все нечётные элементы заменяются на нули. Например, для исходного массива из восьми элементов:

7 1 3 2 14 5 9 6

программа должна вывести (по одному числу в строке) числа

0 0 0 2 14 0 0 6

Паскаль	Алгоритмический язык
<pre>const N = 200; var a: array [1..N] of integer; i, j, k, s: integer; begin for i := 1 to N do readln(a[i]); ... end.</pre>	<pre>алг нач цел N = 200 целтаб a[1:N] цел i, j, k, s нц для i от 1 до N ввод a[i] кц ... кон</pre>
C++	Python
<pre>#include <iostream> using namespace std; const int N = 200; int main() { int a[N]; int i, j, k, s; for (i = 0; i < N; i++) cin >> a[i]; ... }</pre>	<pre>#допускается также использовать #две целочисленные переменные #j, k и s a = [] n = 200 for i in range(n): a.append(int(input()))</pre>

<pre>return 0; }</pre>	
------------------------	--

- 72) Дан целочисленный массив из 200 элементов, в котором записаны значения последовательных результатов измерений. Элементы массива могут принимать целые значения от 0 до 10 000 включительно. Опишите на одном из языков программирования алгоритм, который выполняет выборку данных в массиве: если количество чётных элементов меньше, чем количество нечётных элементов, все чётные элементы заменяются на максимальный из них. Иначе все нечётные элементы заменяются на минимальный из них. Например, для исходного массива из восьми элементов:

7 1 3 2 14 5 9 6

программа должна вывести (по одному числу в строке) числа

7 1 3 14 14 5 9 14

Паскаль	Алгоритмический язык
<pre>const N = 200; var a: array [1..N] of integer; i, j, k, s: integer; begin for i := 1 to N do readln(a[i]); ... end.</pre>	<pre>алг нач цел N = 200 целтаб a[1:N] цел i, j, k, s нц для i от 1 до N ввод a[i] кц ... кон</pre>
C++	Python
<pre>#include <iostream> using namespace std; const int N = 200; int main() { int a[N]; int i, j, k, s; for (i = 0; i < N; i++) cin >> a[i]; ... return 0; }</pre>	<pre>#допускается также использовать #две целочисленные переменные #j, k и s a = [] n = 200 for i in range(n): a.append(int(input()))</pre>

- 73) Дан целочисленный массив из 200 элементов, в котором записаны значения последовательных результатов измерений. Элементы массива могут принимать целые значения от 0 до 10 000 включительно. Опишите на одном из языков программирования алгоритм, который выполняет выборку данных в массиве: если элементов, заканчивающихся на цифру 3, меньше, чем элементов, заканчивающихся на цифру 5, все элементы, заканчивающиеся на цифру 3 заменяются на минимальный из них. Иначе все элементы, заканчивающиеся на цифру 5, заменяются на максимальный из них. Например, для исходного массива из восьми элементов:

15 13 3 27 145 5 93 65

программа должна вывести (по одному числу в строке) числа

15 3 3 27 145 5 3 65

Паскаль	Алгоритмический язык
<pre>const N = 200; var a: array [1..N] of integer; i, j, k, s: integer;</pre>	<pre>алг нач цел N = 200 целтаб a[1:N]</pre>

<pre>begin for i := 1 to N do readln(a[i]); ... end.</pre>	<pre>цел i, j, k, s нц для i от 1 до N ввод a[i] кц ... кон</pre>
C++	Python
<pre>#include <iostream> using namespace std; const int N = 200; int main() { int a[N]; int i, j, k, s; for (i = 0; i < N; i++) cin >> a[i]; ... return 0; }</pre>	<pre>#допускается также использовать #две целочисленные переменные #j, k и s a = [] n = 200 for i in range(n): a.append(int(input()))</pre>