1. 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.

Исходные данные объявлены так, как показано ниже на примерах для некоторых языков программирования. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать некоторые из описанных переменных.

Бейсик

```
CONST N = 20
DIM A (1 TO N) AS
INTEGER
DIM I, J, K AS
INTEGER
FOR I = 1 TO N
INPUT A(I)
NEXT I
…
END
```

Python

```
// допускается
также использовать
// две целочислен-
ные переменные
// j и k
a = []
n = 20
for i in range(0,
n):
a.append(int(input()))
…
```

Паскаль

```
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.
```

Алгоритмический язык

```
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
```

Си++

```
#include
#define N 20
int main() {
int a[N];
int i, j, k;
for (i = 0; i < N; i++)
cin >> a[i];
…
return 0;
}
```

В качестве ответа Вам необходимо привести фрагмент программы, который должен находиться на месте многоточия. Вы можете записать решение также на другом языке программирования (укажите название и используемую версию языка программирования, например, Free Pascal 2.6). В этом случае Вы должны использовать те же самые исходные данные и переменные, какие были предложены в приведённых фрагментах.

2. 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 3, но не делится на 9. В данной задаче под парой подразумеваются два соседних элемента массива.

Исходные данные объявлены так, как показано ниже на примерах для некоторых языков программирования. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать некоторые из описанных переменных.

Бейсик

```
      CONST N = 20
DIM A (1 TO N) AS INTEGER
  DIM I, J, K AS INTEGER
     FOR I = 1 TO N
         INPUT A(I)
        NEXT I
          …
         END
```

Python

```
 // допускается также ис-
        пользовать
// две целочисленные пере-
          менные
         // j и k
          a = []
          n = 20
   for i in range(0, n):
    a.append(int(input()))
              …
```

Паскаль

```
       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.
```

Алгоритмический язык

```
        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
```

Си++

```
  #include <iostream>
     #define N 20
  using namespace std;
     int main() {
       int a[N];
     int i, j, k;
for (i = 0; i < N; i++)
     cin >> a[i];
           …
       return 0;
           }
```

В качестве ответа Вам необходимо привести фрагмент программы, который должен находиться на месте многоточия. Вы можете записать решение также на другом языке программирования (укажите название и используемую версию языка программирования, например, Free Pascal 2.6). В этом случае Вы должны использовать те же самые исходные данные и переменные, какие были предложены в приведённых фрагментах.

3. 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. Исходные данные объявлены так, как показано ниже на примерах для некоторых языков программирования и естественного языка. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать некоторые из описанных переменных.

Бейсик

```
N = 40
DIM A(N) AS LONG
DIM I, J, K AS
LONG
FOR I = 1 TO N
INPUT A(I)
NEXT I
...
END
```

Паскаль

```
const
N = 40;
var
a: array [1..N]
of longint;
i, j, k: longint;
begin
for i := 1 to N
do
readln(a[i]);
...
end.
```

Си++

```
#include
#define N 40
int main()
{
long a[N];
long i, j, k;
for (i = 0; i <
N; i++)
cin >> a[i];
...
}
```

Алгоритмический язык

```
алг
нач
цел N = 40
цел таб a[1:N]
цел i, j, k
нц для i от 1 до
N
ввод a[i]
кц
...
кон
```

Естественный язык

Объявляем массив A из 40 элементов.
Объявляем целочисленные переменные I, J, K.
В цикле от 1 до 40 вводим элементы массива A с 1-го по 40-й.
...

Python

```
// допускается также использовать
// две целочисленные переменные
// j и k
a = []
n = 40
for i in range(0, n):
a.append(int(input()))
```

В качестве ответа Вам необходимо привести фрагмент программы, который должен находиться на месте многоточия. Вы можете записать решение также на другом языке программирования (укажите название и используемую версию языка программирования, например, Free Pascal 2.6). В этом случае Вы должны использовать те же самые исходные данные и переменные, какие были предложены в приведённых фрагментах.

4. 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. Исходные данные объявлены так, как показано ниже на примерах для некоторых языков программирования и естественного языка. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать некоторые из описанных переменных.

Бейсик

```
N = 40
DIM A(N) AS LONG
DIM I, J, K AS
LONG
FOR I = 1 TO N
INPUT A(I)
NEXT I
...
END
```

Паскаль

```
const
N = 40;
var
a: array [1..N]
of longint;
i, j, k: longint;
begin
for i := 1 to N
do
readln(a[i]);
...
end.
```

Си++

```
#include
<iostream>
using namespace
std;
#define N 40
int main()
{
long a[N];
long i, j, k;
for (i = 0; i <
N; i++)
cin >> a[i];
...
}
```

Алгоритмический язык

```
алг
нач
цел N = 40
цел таб a[1:N]
цел i, j, k
нц для i от 1 до
N
ввод a[i]
кц
...
кон
```

Естественный язык

Объявляем массив A из 40 элементов.
Объявляем целочисленные переменные I, J, K.
В цикле от 1 до 40 вводим элементы массива A с 1-го по 40-й.

Python

```
# допускается также
# использовать две
# целочисленные переменные j, k
a = []
n = 40
for i in range(0, n):
a.append(int(input()))
...
```

В качестве ответа Вам необходимо привести фрагмент программы, который должен находиться на месте многоточия. Вы можете записать решение также на другом языке программирования (укажите название и используемую версию языка программирования, например, Free Pascal 2.6). В этом случае Вы должны использовать те же самые исходные данные и переменные, какие были предложены в приведённых фрагментах.

5. 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; 9; –3; 6 – ответ: 2.

Исходные данные объявлены так, как показано ниже на примерах для некоторых языков программирования и естественного языка. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать некоторые из описанных переменных.

Бейсик

```
CONST N AS
INTEGER = 20
DIM A (1 TO N) AS
INTEGER
DIM I AS INTEGER,
J AS INTEGER,
K AS INTEGER
FOR I = 1 TO N
INPUT A(I)
NEXT I
...

END
```

Паскаль

```
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.
```

Си++

```
#include
<iostream>
using namespace
std;
#define N 20
int main() {
int a[N];
int i, j, k;
for (i = 0; i
< N; i++)
cin >>
a[i];
...
return 0;
}
```

Алгоритмический язык

```
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
```

Python

```
# допускается
также
# использовать две
# целочисленные
переменные j и k
a = []
n = 20
for i in range(0,
n):
a.append(int(input()))
...
```

Естественный язык

Объявляем массив A из 20 элементов.
Объявляем целочисленные переменные I, J, K.
В цикле от 1 до 20 вводим элементы массива A с 1-го по 20-й.
…

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 Free Pascal 2.6) 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).

6. 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, сумма которых чётна, а произведение больше 100. Под парой подразумевается два подряд идущих элемента массива. Исходные данные объявлены так, как показано ниже на примерах для некоторых языков программирования и естественного языка. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать некоторые из описанных переменных.

Бейсик

```
N = 40
DIM A(N) AS LONG
DIM I, J, K AS LONG
FOR I = 1 TO N
    INPUT A(I)
NEXT I
...
END
```

Паскаль

```
const
N = 40;
var
a: array [1..N] of longint;
i, j, k: longint;
begin
    for i := 1 to N do
readln(a[i]);
...
end.
```

Си++

```
#include <iostream>
using namespace std;
#define N 40
int main()
{
    long a[N];
    long i, j, k;
    for (i=0; i<N; i++)
    cin >> a[i];
    ...
}
```

Естественный язык

Объявляем массив A из 40 элементов.
Объявляем целочисленные переменные I, J, K.
В цикле от 1 до 40 вводим элементы массива
A с 1-го по 40-й.
...

Алгоритмический язык

```
алг
нач
    цел N = 40
    цел таб a[1:N]
    цел i, j, k
    нц для i от 1 до N
        ввод a[i]
    кц
    ...
кон
```

Python

```
N = 40
i = None
j = None
k = None
a  =[int(input())  for  i  in
range(N)]
...
```

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 Free Pascal 2.6) 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).

7. 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, сумма которых нечётна, а произведение меньше 100. Под парой подразумевается два подряд идущих элемента массива. Исходные данные объявлены так, как показано ниже на примерах для некоторых языков программирования и естественного языка. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать некоторые из описанных переменных.

Бейсик

```
N = 40
DIM A(N) AS LONG
DIM I, J, K AS LONG
FOR I = 1 TO N
    INPUT A(I)
NEXT I
...
END
```

Паскаль

```
const
N = 40;
var
a: array [1..N] of longint;
i, j, k: longint;
begin
    for i := 1 to N do
readln(a[i]);
...
end.
```

Си++

```
#include <iostream>
using namespace std;
#define N 40
int main()
{
    long a[N];
    long i, j, k;
    for (i=0; i<N; i++)
    cin >> a[i];
    ...
}
```

Естественный язык

Объявляем массив A из 40 элементов.
Объявляем целочисленные переменные I, J, K.
В цикле от 1 до 40 вводим элементы массива
A с 1-го по 40-й.
...

Алгоритмический язык

```
алг
нач
    цел N = 40
    цел таб a[1:N]
    цел i, j, k
    нц для i от 1 до N
        ввод a[i]
    кц
    ...
кон
```

Python

```
N = 40
i = None
j = None
k = None
a  =[int(input())  for  i  in
range(N)]
...
```

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 Free Pascal 2.6) 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).

8. 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. Например, для массива из пяти элементов: 6; 2; 9; –3; 6 – ответ: 4.

Исходные данные объявлены так, как показано ниже на примерах для некоторых языков программирования и естественного языка. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать некоторые из описанных переменных.

Бейсик

```
CONST N AS
INTEGER = 20
DIM A (1 TO N) AS
INTEGER
DIM I AS INTEGER,
J AS
INTEGER,
K AS
INTEGER

FOR I = 1 TO N
INPUT A(I)
NEXT I
...
END
```

Python

```
# допускается
также
# использовать две
# целочисленные
переменные j и k
a = []
n = 20
for i in range(0,
n):
a.append(int(input()))
...
```

Паскаль

```
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.
```

Алгоритмический язык

```
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
```

Си++

```
#include
<iostream>
using namespace
std;
#define N 20
int main() {
int a[N];
int i, j, k;
for (i = 0; i
< N; i++)
cin >>
a[i];
...
return 0;
}
```

Естественный язык

Объявляем массив A из 20 элементов.
Объявляем целочисленные переменные I, J, K.
В цикле от 1 до 20 вводим элементы массива A с 1-го по 20-й.
…

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 Free Pascal 2.6) 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).

9. 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 на примерах для некоторых языков программирования и естественного языка. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать некоторые из описанных переменных.

Бейсик

```
N = 20
DIM A(N) AS
INTEGER
DIM I, J, K AS
INTEGER
FOR I = 1 TO N
INPUT A(I)
NEXT I
...
END
```

Python

```
N = 20
i = None
j = None
k = None
a = [int(input())
for i in
range(N)]
...
```

Паскаль

```
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.
```

Алгоритмический язык

```
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
```

Си++

```
#include
<iostream>
using namespace
std;
int main(void)
{
const int N = 20;
int a[N];
int i, j, k;
for (i = 0; i <
N; i++)
cin >> a[i];
...
}
```

Естественный язык

Объявляем массив A из 20 элементов.
Объявляем целочисленные переменные I, J, K.
В цикле от 1 до 20 вводим элементы массива A с 1-го по 20-й.
...

10. Дан массив, содержащий 2014 вещественных чисел. Напишите на одном из языков программирования программу, находящую в этом массиве два соседних элемента, значения которых наименее близки, то есть абсолютная величина их разности максимальна. Если таких пар несколько, можно взять любую из них. Программа должна вывести найденные элементы. Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать часть из описанных.

Бейсик

```
      N=2014
  DIM A(N) AS SINGLE
  DIM D, R AS SINGLE
    DIM I, J, K AS
       INTEGER
    FOR I = 1 TO N
      INPUT A(I)
       NEXT I
         …
        END
```

Паскаль

```
        const
       N=2014;
         var
  a: array [1..N] of
        real;
     d, r: real;
   i, j, k: integer;
        begin
   for i:=1 to N do
     readln(a[i]);
          …
        end.
```

Си++

```
 #include <iostream>
using namespace std;
 #include <math.h>
  #define N 2014
    int main(){
    float a[N];
    float d, r;
   int i, j, k;
for (i=0; i < N; i++)
    cin >> a[i];
         …
         }
```

Алгоритмический язык

```
       алг
       нач
    цел N=2014
  вещтаб a[1:N]
    вещ d, r;
   цел i, j, k
нц для i от 1 до N
    ввод a[i]
       кц
       …
       кон
```

Python

```
      # допускается также
       # использовать две
# целочисленные переменные j, k
     # и вещественные d, r
            a = []
          n = 2014
     for i in range(0, n):
      a.append(float(input()))
              ...
```

В качестве ответа Вам необходимо привести фрагмент программы, который должен находиться на месте многоточия. Вы можете записать решение также на другом языке программирования (укажите название и используемую версию языка программирования, например Free Pascal 2.4). В этом случае вы должны использовать те же самые исходные данные и переменные, какие были предложены в условии.

11. Дан целочисленный массив из 30 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 13. В данной задаче под парой подразумевается два подряд идущих элемента массива. Например, для массива из пяти элементов: 13; 7; 26; −1; 9 — ответ: 3. Исходные данные объявлены так, как показано ниже на примерах для некоторых языков программирования и естественного языка. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать некоторые из описанных переменных.

Бейсик

```
CONST N AS
INTEGER = 30
DIM A (1 TO N) AS
INTEGER
DIM I AS INTEGER,
J AS INTEGER,
K AS INTEGER
FOR I = 1 TO N
INPUT A(I)
NEXT I
...
END
```

Паскаль

```
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.
```

Си++

```
#include
<iostream>
using namespace
std;
#define N 30
int main() {
int a[N];
int i, j, k;
for (i = 0; i
< N; i++)
cin >>
a[i];
...
return 0;
}
```

Алгоритмический язык

```
алг
нач
цел N = 30
целтаб a[1:N]
цел i, j, k
нц для i от 1
до N
ввод a[i]
кц
...

кон
```

Python

```
# допускается
также
# использовать две
# целочисленные
переменные j и k
a = []
n = 30
for i in range(0,
n):
a.append(int(input()))
...
```

Естественный язык

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.
…

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 Free Pascal 2.6) 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).

12. Дан целочисленный массив из 30 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 не делится на 7. В данной задаче под парой подразумевается два подряд идущих элемента массива. Например, для массива из пяти элементов: 13; 7; 26; -1; 9 — ответ: 4.

Исходные данные объявлены так, как показано ниже на примерах для некоторых языков программирования и естественного языка. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать некоторые из описанных переменных.

Бейсик

```
CONST N AS
INTEGER = 30
DIM A (1 TO N) AS
INTEGER
DIM I AS INTEGER,
J AS INTEGER,
K AS INTEGER
FOR I = 1 TO N
INPUT A(I)
NEXT I
...
END
```

Паскаль

```
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.
```

Си

```
#include
<iostream>
using namespace
std;
#define N 30
int main() {
int a[N];
int i, j, k;
for (i = 0; i
< N; i++)
cin >>
a[i];
...
return 0;
}
```

Алгоритмический язык

```
алг
нач
цел N = 30
целтаб a[1:N]
цел i, j, k
нц для i от 1
до N
ввод a[i]
кц
...

кон
```

Python

```
# допускается
также
# использовать две
# целочисленные
переменные j и k
a = []
n = 30
for i in range(0,
n):
a.append(int(input()))
...
```

Естественный язык

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.
…

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 Free Pascal 2.6) 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 50 элементов. Элементы массива могут принимать целые значения от 0 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 двузначные. В данной задаче под парой подразумевается два подряд идущих элемента массива.

Например, для массива из пяти элементов: 16 2 14 91 21 — ответ: 2. Исходные данные объявлены так, как показано ниже на примерах для некоторых языков программирования и естественного языка. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать некоторые из описанных переменных.

Бейсик

```
CONST N = 50
DIM A (1 TO N) AS
INTEGER
DIM I, J, K, AS
INTEGER
FOR I = 1 TO N
INPUT A(I)
NEXT I
...
END
```

Python

```
//допускается
также использовать
//две целочислен-
ные переменные j и
k
a = []
n = 50
for i in range(0,
n):
a.append(int(input()))
...
```

Алгоритмический язык

```
алг
нач
цел N = 50
целтаб a[1:N]
цел i, j, k
нц для i от 1
до N
ввод a[i]
кц
...
кон
```

Паскаль

```
const
N = 50;
var
a: array
[1..N] of integer;
i, j, k:
integer;
begin
for i := 1 to
N do
readln(a[i]);
...
end.
```

Си++

```
#include
<iostream>
using namespace
std;
#define N 50
int main() {
int a[N];
int i, j, k;
for (i = 0;
i<N; i++)
cin >>
a[i];
...
return 0;
}
```

Естественный язык

Объявляем массив A из 50 элементов.
Объявляем целочисленные переменные I, J, K.
В цикле от 1 до 50 вводим элементы массива A с 1-го по 50-й.
…

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, Free Pascal 2.6) 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 массив из 2000 элементов. Хорошей парой называется пара элементов, из которых оба оканчиваются на 9, при этом элементы являются соседями. Посчитайте количество хороших пар. Каждый элемент по модулю не превышает 30 000. В ответе укажите кусок программы на месте многоточия. Разрешено не использовать некоторые из инициализированных переменных, но запрещено пользоваться переменными, не указанными ниже.

Бейсик

```
CONST N = 2000
DIM A (1 TO N) AS
    INTEGER
DIM I, J, K AS
    INTEGER
FOR I = 1 TO N
  INPUT A(I)
  NEXT I
    …
  END
```

Python

```
  // допускается
также использовать
// две целочислен-
  ные переменные
     // j и k
      a = []
     n = 2000
for i in range(0,
        n):
a.append(int(input()))
        …
```

Паскаль

```
     const
   N = 2000;
     var
 a: array [1..N]
      of
    integer;
i, j, k: integer;
     begin
 for i := 1 to N
      do
  readln(a[i]);
       …
     end.
```

Алгоритмический язык

```
       алг
       нач
  цел N = 2000
  целтаб a[1:N]
   цел i, j, k
нц для i от 1 до
       N
   ввод a[i]
       кц
       …
       кон
```

Си++

```
      #include
   #define N 2000
    int main() {
      int a[N];
    int i, j, k;
for (i = 0; i < N; i++)
    cin >> a[i];
         …
      return 0;
          }
```

15. Дан массив. Необходимо найти количество соседних пар элементов таких, что оба числа из пары делятся нацело на 4. Объявлены переменные N, k, i, массив a из N элементов.

16. Дается последовательность чисел. Нужно определить, сколько есть пар чисел, в которых есть хотя бы одно число, оканчивающееся на "3".

В данной задаче под парой подразумевается два подряд идущих элемента массива.

17. Дан целочисленный массив из 30 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 обеих чисел оканчивается на 6.
Под парами понимайте два смежных, соседних элемента массива.

18. Дан целочисленный массив из 40 элементов. Элементы массива могут принимать целые значения от 0 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 десятичная запись хотя бы одного числа оканчивается на 2. В данной задаче под парой подразумевается два подряд идущих элемента массива.

Например, для массива из пяти элементов: 16 3 142 55 22 – ответ: 3.

Исходные данные объявлены так, как показано ниже на примерах для некоторых языков программирования и естественного языка. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать некоторые из описанных переменных.

Бейсик

```
CONST N = 40
DIM A (1 TO N) AS
    INTEGER
 DIM I, J, K, AS
    INTEGER

 FOR I = 1 TO N
    INPUT A(I)
    NEXT I
      ...
     END
```

Python

```
#допускается также
   использовать
#две целочисленные
 переменные j и k
      a = []
      n = 40
 for i in range(0,
       n):
a.append(int(input()))
        ...
```

Паскаль

```
      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.
```

Алгоритмический язык

```
        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
```

Си++

```
    #include
   <iostream>
 using namespace
      std;
  #define N 40
  int main() {
     int a[N];
    int i, j, k;
   for (i = 0; i
```

Естественный язык

`< N; i++)`
`cin >>`
`a[i];`
`...`
`return 0;`
`}`
	Объявляем массив *A* из 40 элементов.
	Объявляем целочисленные переменные *I, J, K*.
	В цикле от 1 до 40 вводим элементы массива *A* с 1-го по 40-й.
	…

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, Free Pascal 2.6) 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 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 13. В данной задаче под парой подразумевается два подряд идущих элемента массива. Например, для массива из пяти элементов: 6; 2; 13; –26; 14 — ответ: 3.

Исходные данные объявлены так, как показано ниже на примерах для некоторых языков программирования и естественного языка. Запрещается использовать переменные, не описанные ниже, но разрешается
не использовать некоторые из описанных переменных.

Бейсик

```
CONST N AS
INTEGER = 20
DIM A (1 TO N) AS
INTEGER
DIM I AS INTEGER,
J AS
INTEGER,
K AS
INTEGER

FOR I = 1 TO N
INPUT A(I)
NEXT I
...
END
```

Python

```
# допускается
также
# использовать две
# целочисленные
переменные j и k
a = []
n = 20
for i in range(0,
n):
a.append(int(input()))
...
```

Паскаль

```
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.
```

Алгоритмический язык

```
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
```

Си

```
#include
#define N 20
int main() {
int a[N];
int i, j, k;
for (i = 0; i
< N; i++)
cin >>
a[i];
...
return 0;
}
```

Естественный язык

Объявляем массив A из 20 элементов.
Объявляем целочисленные переменные I, J, K.
В цикле от 1 до 20 вводим элементы массива A с 1-го по 20-й.
…

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 Free Pascal 2.6) 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 20 элементов. Элементы массива могут принимать целые значения от 0 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 являются чётными. В данной задаче под парой подразумевается два подряд идущих элемента массива.

Например, для массива из пяти элементов: 6; 1; 4; 6; 10 – ответ: 2. Исходные данные объявлены так, как показано ниже на примерах для некоторых языков программирования и естественного языка. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать некоторые из описанных переменных.

Бейсик

```
CONST N AS
INTEGER = 20
DIM A (1 TO N) AS
INTEGER
DIM I AS INTEGER,
J AS INTEGER,
K AS INTEGER
FOR I = 1 TO N
INPUT A(I)
NEXT I
...
END
```

Python

```
# допускается
также
# использовать две
# целочисленные
переменные j и k
a = []
n = 20
for i in range(0,
n):
a.append(int(input()))
```

Паскаль

```
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.
```

Алгоритмический язык

```
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
```

Си++

```
#include
<iostream>
using namespace
std;
#define N 20
int main() {
int a[N];
int i, j, k;
for (i = 0; i
< N; i++)
cin >>
a[i];
... return 0;
}
```

Естественный язык

Объявляем массив A из 20 элементов.
Объявляем целочисленные переменные I, J, K.
В цикле от 1 до 20 вводим элементы массива A
с 1-го по 20-й.
…

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 Free Pascal 2.6) 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).