

# **Основные подходы к решению заданий ЕГЭ по информатике**

**Председатель ОМО учителей информатики,  
учитель МКОУ «Новодугинская СШ»  
Иванова Наталья Михайловна**

**2022 г.**

Наименование программного обеспечения	Версия программного обеспечения	Ссылка на ПО
C++	Microsoft Visual Studio 2019 CodeBlocks 20.03+ mingw	<a href="https://visualstudio.microsoft.com/ru/vs/community/">https://visualstudio.microsoft.com/ru/vs/community/</a> <a href="https://www.codeblocks.org/downloads/binaries/">https://www.codeblocks.org/downloads/binaries/</a>
Java	Java 11 и IntelliJ IDEA Community Edition	<a href="https://www.jetbrains.com/ru-ru/idea/download/download-thanks.html?platform=windows&amp;code=IC">https://www.jetbrains.com/ru-ru/idea/download/download-thanks.html?platform=windows&amp;code=IC</a>
C#	Microsoft Visual Studio 2019	<a href="https://visualstudio.microsoft.com/ru/vs/community/">https://visualstudio.microsoft.com/ru/vs/community/</a>
Pascal	PascalABC.NET (версия 3.8, сборка 2857 от 07.03.2021)	<a href="http://pascalabc.net/ssyilki-dlya-skachivaniya">http://pascalabc.net/ssyilki-dlya-skachivaniya</a>
Python	Python 3.8.8 PyCharm Community Edition 2020.3.2 x64 Wing 101	<a href="https://www.python.org/downloads/">https://www.python.org/downloads/</a> <a href="https://www.jetbrains.com/ru-ru/pycharm/download/#section=windows">https://www.jetbrains.com/ru-ru/pycharm/download/#section=windows</a> <a href="https://wingware.com/downloads/wing-101">https://wingware.com/downloads/wing-101</a>
Школьный алгоритмический язык	КуМир 2.1.0	<a href="https://www.niisi.ru/kumir/dl.htm">https://www.niisi.ru/kumir/dl.htm</a>
Текстовый редактор, редактор таблиц и презентаций	P7-Офис 6.1.0.68  LibreOffice 7.3.0	x64: <a href="https://cloud.lad-soft.ru/index.php/s/DzQ4mBAtJDfJocw">https://cloud.lad-soft.ru/index.php/s/DzQ4mBAtJDfJocw</a> x86: <a href="https://cloud.lad-soft.ru/index.php/s/2wGNezjG4XJEz4i">https://cloud.lad-soft.ru/index.php/s/2wGNezjG4XJEz4i</a> <a href="https://www.libreoffice.org/download/download/?type=win-x86_64&amp;version=6.4.7&amp;lang=ru">https://www.libreoffice.org/download/download/?type=win-x86_64&amp;version=6.4.7&amp;lang=ru</a>

# Дополнительные материалы

[Инструкция для участника КЕГЭ по использованию ПО.pdf](#) (956 КБ)

[ЧЕРНОВИК КЕГЭ 2022.pdf](#) (468 КБ)

# Задание 5

На вход алгоритма подаётся натуральное число  $N$ . Алгоритм строит по нему новое число  $R$  следующим образом.

1. Строится двоичная запись числа  $N$ .
2. К этой записи дописываются справа ещё два разряда по следующему правилу:
  - а) складываются все цифры двоичной записи числа  $N$ , и остаток от деления суммы на 2 дописывается в конец числа (справа). Например, запись 11100 преобразуется в запись 111001;
  - б) над этой записью производятся те же действия – справа дописывается остаток от деления суммы её цифр на 2.

Полученная таким образом запись (в ней на два разряда больше, чем в записи исходного числа  $N$ ) является двоичной записью искомого числа  $R$ . Укажите такое наименьшее число  $N$ , для которого результат работы данного алгоритма больше числа 77. В ответе это число запишите в десятичной системе счисления.

## Решение (с помощью Калькулятора Windows)

- 1) переключаем Калькулятор в режим Программист (Вид – Программист или Alt+3);
- 2) в десятичной системе (по умолчанию включен режим *Dec*) набираем 78;
- 3) под окошком вывода отображается двоичный код 78 (0...01001110);
- 4) т.к. двоичный код содержит четное количество единиц (четыре), то  $R$  может равняться 78.
- 5) чтобы получить ответ ( $N$ ) надо от двоичного кода  $R=78$  отбросить два правых разряда. Для этого можно использовать команду Калькулятора сдвиг вправо (*Right Shift*): нажать кнопку *Rsh*, затем кнопку «2» (сдвиг на два разряда) и кнопку «=»;
- 6) в окошке вывода видим ответ в десятичном коде: 19

Ответ: 19

```
uses school;
var n,i,k,a,b,c: integer;
    s: string;
begin
  for n:= 1 to 70 do begin
    s:= bin(n);
    k:= 0;
    for i:=1 to s.Length do
      k += StrToInt(s[i]);
    s+= k mod 2;
    k:= 0;
    for i:=1 to s.Length do
      k += StrToInt(s[i]);
    s+= k mod 2;
    c:= dec(s,2);
    if c > 77 then begin
      println(n);
      break
    end;
  end;
end.
```

Ответ: 19.

# Задание 5

5

На вход алгоритма подаётся натуральное число  $N$ . Алгоритм строит по нему новое число  $R$  следующим образом.

1. Строится двоичная запись числа  $N$ .
2. Каждый разряд этой записи заменяется двумя разрядами по следующему правилу: если в разряде стоит 0, то вместо него пишется 00; если в разряде стоит 1, то 1 заменяется на 11.

Например, двоичная запись 1001 числа 9 будет преобразована в 11000011.

Полученная таким образом запись (в ней в два раза больше разрядов, чем в записи исходного числа  $N$ ) является двоичной записью числа  $R$  — результата работы данного алгоритма.

Укажите минимальное число  $R$ , большее 32, которое может являться результатом работы данного алгоритма. В ответе это число запишите в десятичной системе счисления.

```
for n in range(0, 100):
    s = bin(n)[2::]
    s1=""
    for i in range(0,len(s)):
        if s[i]=='0':
            s1+='00'
        else:
            s1+='11'
    r = int(s1, base = 2)
    if r >= 32:
        print(n,r)
```

# ЗАДАНИЕ 6

Определите, при каком наименьшем введённом значении переменной  $s$  программа выведет число 64.

```
var s, n: integer;
begin
  readln (s);
  n := 1;
  while s < 51 do begin
    s := s + 5;
    n := n * 2
  end;
  writeln(n)
end.
```

Решение (теоретическое):

- 1) из программы видно, что в конце программы выводится значение переменной  $n$
- 2) из программы видно, что начальное значение переменной  $n$  равно 1
- 3) на каждой итерации цикла значение переменной  $n$  увеличивается в 2 раза
- 4) поскольку  $64 = 2^6$ , для того чтобы получить  $n=64$ , необходимо выполнить тело цикла 6 раз
- 5) при каждой итерации цикла значение переменной  $s$  увеличивается на 5, то есть после 6 итераций мы получим  $s=s_0+6*5=s_0+30$ , где  $s_0$  – введённое начальное значение  $s$
- 6) цикл останавливается при условии  $s \geq 51$ , то есть при  $s_0+30 \geq 51$  или  $s_0 \geq 21$
- 7) Ответ: **21**.

```
def f( s ):
  n = 1
  while s < 51:
    s = s + 5
    n = n * 2
  return n

s = 50
while True:
  if f(s) == 64: print(s)
  s -= 1
```

# Задание 6

6 Определите, при каком **наибольшем** введённом значении переменной  $n$  программа выведет число 627. Для Вашего удобства программа представлена на четырёх языках программирования.

Python	Алгоритмический язык
<pre>s = 127 n = int(input()) while n - s &gt; 0:     s = s + 20     n = n + 15 print(s)</pre>	<pre>алг нач   цел n, s   s := 127   ввод n   <u>нц пока</u> n - s &gt; 0     s := s + 20     n := n + 15   <u>кц</u>   вывод s кон</pre>

```
def DMAX(n):
    for i in range(n//2+1,1,-1):
        if n % i == 0:
            return i
    return 0
```

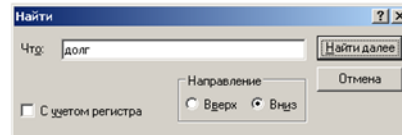
```
print(DMAX(175001))
k = 0
x = 450001
while k <= 5:
    m = DMAX(DMAX(x))
    if (m != 0):
        k+=1
        print(x, DMAX(x))
    x += 1
```

# Задание 10

С помощью текстового редактора определите, сколько раз, не считая сносок, встречается слово «долг» или «Долг» в тексте романа в стихах А.С. Пушкина «Евгений Онегин» (файлы 10-0.docx, 10-0.txt). Другие формы слова «долг», такие как «долги», «долгами» и т.д., учитывать не следует. В ответе укажите только число.

## Решение (простейшие текстовые редакторы):

- 1) Решение этой задачи существенно зависит от возможностей программы, которую вы используете. Рассмотрим сначала самый худший вариант, когда в вашем распоряжении только простейший текстовый редактор типа Блокнота.
- 2) Загрузите файл 10-0.txt в редактор.
- 3) С помощью комбинации клавиш *Ctrl+F* (или верхнего меню *Правка – Найти*) нужно вызвать окно поиска и ввести нужное слово:



- 4) **Важно!** Поскольку нас интересует слово «долг», которое может начинаться как с заглавной, так и со строчной буквы, НЕ нужно включать флажок «С учетом регистра»
- 5) далее щелкаем по кнопке *Найти далее* (можно также щёлкнуть мышью на тексте и нажимать клавишу *F3*)
- 6) к сожалению, простые программы ищут только заданную цепочку символов и не умеют искать слово в одной заданной форме (то есть будут найдены также слова «долго», «долги», «долгий» и т.п., которые нас не интересуют); поэтому приходится просматривать все найденные слова и вручную считать, сколько раз встретится именно слово «долг», а не другие слова, содержащие эту цепочку символов
- 7) Ответ: **2**.

- 1) можно использовать программу на Python, которая читает в память файл целиком, разбивает его на слова, переводит все символы в нижний регистр (к строчным буквам) и считает, сколько заданных слов оказалось в этом списке:

```
import re
n = 0
for x in re.split('\W+', open('10-0.txt', 'r').read()).lower():
    if x == 'долг':
        n += 1
print(n)
```



# Задание 10

10

С помощью текстового редактора определите, сколько раз, не считая сносок, встречается слово «кучер» или «Кучер» в тексте поэмы Н. В. Гоголя «Мёртвые души»<sup>1</sup>. Другие формы слова «кучер», такие как «кучеру», «кучером» и т. д., учитывать не следует. В ответе укажите только число.

Ответ: \_\_\_\_\_.

```
with
open('C:\\Users\\EGE1\\Desktop\\EGE2022
\\10\\Мертвые души.txt') as f:
    k = 0
    for i in f.readlines():
        for j in i.split():
            if 'кучер' in j or 'Кучер' in j:
                print(j)
                k += 1
print(k)
```

# Задание 12

Редактор получает на вход строку цифр и преобразовывает её. Редактор может выполнять две команды, в обеих командах  $v$  и  $w$  обозначают цепочки цифр.

**заменить ( $v, w$ )**

Эта команда заменяет в строке первое слева вхождение цепочки  $v$  на цепочку  $w$ . Если цепочки  $v$  в строке нет, эта команда не изменяет строку.

**нашлось ( $v$ )**

Эта команда проверяет, встречается ли цепочка  $v$  в строке исполнителя Редактор. Если она встречается, то команда возвращает логическое значение «истина», в противном случае возвращает значение «ложь». Строка при этом не изменяется.

Дана программа для исполнителя Редактор:

НАЧАЛО

ПОКА нашлось (2222) ИЛИ нашлось (8888)

  ЕСЛИ нашлось (2222)

    ТО заменить (2222, 88)

    ИНАЧЕ заменить (8888, 22)

  КОНЕЦ ЕСЛИ

КОНЕЦ ПОКА

КОНЕЦ

Какая строка получится в результате применения приведённой программы к строке, состоящей из 70 идущих подряд цифр 8? В ответе запишите полученную строку.

1) программа на языке PascalABC.NET:

```
begin
  var s := StringOfChar('8', 70);
  var p2 := Pos('2222',s);
  var p8 := Pos('8888',s);
  while (p2 > 0) or (p8 > 0) do begin
    if p2 > 0 then begin
      Delete( s, p2, 4 );
      Insert( '88', s, p2 );
    end
    else begin
      Delete( s, p8, 4 );
      Insert( '22', s, p8 );
    end;
    p2 := Pos('2222',s);
    p8 := Pos('8888',s);
  end;
  write(s);
end.
```

# Задание 12

Определите количество цифр 2 в строке, получившейся в результате применения приведённой ниже программы к входной строке  $\underbrace{1222\dots22}_{70}$ , т. е. к строке, состоящей из единицы, за которой следуют 70 цифр 2 подряд.  
В ответе запишите только количество цифр 2 в получившейся строке.

```
НАЧАЛО
ПОКА нашлось (12) ИЛИ нашлось (1)
  ЕСЛИ нашлось (12)
    ТО заменить (12, 221)
  ИНАЧЕ
    ЕСЛИ нашлось (1)
      ТО заменить (1, 2)
  КОНЕЦ ЕСЛИ
КОНЕЦ ЕСЛИ
КОНЕЦ ПОКА
КОНЕЦ
```

Ответ: \_\_\_\_\_.

```
s = '1'+ '2'*70
while '12' in s or '1' in s:
    if '12' in s:
        s=s.replace('12','221',1)
    elif '1' in s:
        s=s.replace('1','2',1)
print(s,s.count('2'))
```

# Задание 14

Значение арифметического выражения:  $49^7 + 7^{21} - 7$  – записали в системе счисления с основанием 7. Сколько цифр 6 содержится в этой записи?

**Решение:**

- 1) приведём все числа к степеням семерки, учитывая, что  $49 = 7^2$   
 $7^{14} + 7^{21} - 7^1$
- 2) расставим степени в порядке убывания:  
 $7^{21} + 7^{14} - 7^1$
- 3) Очевидно, что «шестёрки» в семеричной записи значения выражения возникнут только за счёт вычисления разности  $7^{14} - 7^1$ , их количество равно  $14 - 1 = 13$
- 4) Ответ: **13**.

```
x = 49**7 + 7**21 - 7
count6 = 0
while x:
    if x % 7 == 6: count6 += 1
    x //= 7
print( count6 )
```

# Задание 14

14

Значение арифметического выражения:  $4^{100} - 4^{65} + 16^{15} - 64$  — записали в системе счисления с основанием 4. Сколько цифр «3» содержится в этой записи?

Ответ: \_\_\_\_\_.

```
n=4**100-4**65+16**15-64
k = 0
while n > 0:
    if n % 4 == 3:
        k += 1
    n = n // 4
print(k)
```

# Задание 15

Обозначим через **ДЕЛ** ( $n, m$ ) утверждение «натуральное число  $n$  делится без остатка на натуральное число  $m$ ». Для какого наибольшего натурального числа  $A$  формула

$$\neg \text{ДЕЛ}(x, A) \rightarrow (\text{ДЕЛ}(x, 6) \rightarrow \neg \text{ДЕЛ}(x, 9))$$

тождественно истинна (то есть принимает значение 1 при любом натуральном значении переменной  $x$ )?

1) для сокращения записи введём обозначения:

$$\text{ДЕЛ}(x, A) = A$$

$$\text{ДЕЛ}(x, 6) = D_6$$

$$\text{ДЕЛ}(x, 9) = D_9$$

2) перепишем выражение в виде  $\bar{A} \rightarrow (D_6 \rightarrow \bar{D}_9) = 1$

3) используя формулу  $A \rightarrow B = \bar{A} + B$ , раскроем первую импликацию:

$$A + (D_6 \rightarrow \bar{D}_9) = 1$$

4) и вторую:

$$A + \bar{D}_6 + \bar{D}_9 = 1$$

5) согласно правилу де Моргана  $\bar{D}_6 + \bar{D}_9 = \overline{D_6 \cdot D_9}$ , так что

$$A + \overline{D_6 \cdot D_9} = 1$$

6) сведём выражение к единственной импликации

$$D_6 \cdot D_9 \rightarrow A = 1$$

7) сформулируем правило, которое мы получили: если значение  $x$  делится на 6 и делится на 9, то оно делится на  $A$ ;

8) если значение  $x$  делится на 6 и делится на 9, то оно делится на наименьшее общее кратное  $\text{НОК}(6,9)=18$ , поэтому наибольшее значение  $A$ , удовлетворяющее условию, равно 18

1) <https://www.youtube.com/watch?v=vdlIelsomkM>:

```
for A in range(1,1000):
    OK = 1
    for x in range(1,1000):
        OK *= (x % A != 0) <= ((x % 6 == 0) <= (x % 9 != 0))
    if OK:
print( A
```

# Задание 15

15

Для какого наибольшего целого неотрицательного числа  $A$  выражение

$$(x < A) \wedge (y < A) \wedge (x \cdot y > 601)$$

тождественно ложно, т. е. принимает значение 0 при любых целых положительных  $x$  и  $y$ ?

Ответ: \_\_\_\_\_.

```
for a in range(100,0,-1):
    flag = True
    for x in range(0,500):
        for y in range(0,500):
            if ((x < a) and (y < a) and (x*y >
601)) == True:
                flag = False
    if flag == True:
        print(a)
        break
```

# Задание 16

Алгоритм вычисления функции  $F(n)$  задан следующими соотношениями:

$$F(n) = 1 \text{ при } n = 1$$

$$F(n) = n + F(n-1), \text{ если } n \text{ чётно,}$$

$$F(n) = 2 \cdot F(n-2), \text{ если } n > 1 \text{ и } n \text{ нечётно.}$$

Чему равно значение функции  $F(26)$ ?

**Решение (ручной счёт от последнего значения):**

1) чтобы вычислить  $F(26)$ , используем формулу для чётных  $n$ :

$$F(26) = 26 + F(25)$$

2) нам неизвестно значение  $F(25)$ , поэтому, применяя формулу для нечётных  $n$ , находим

$$F(25) = 2 \cdot F(23)$$

3) далее так же придётся написать формулы для вычисления  $F(23)$ ,  $F(21)$ , ...,  $F(3)$ , и в конце мы получим

$$F(3) = 2 \cdot F(1)$$

4) значение  $F(1) = 1$  нам задано, подставляя его в предыдущую формулу, находим

$$F(3) = 2 \cdot F(1) = 2$$

5) теперь значение подставляем в формулу для  $F(5)$ , потом найденное значение  $F(5)$  – в формулу для  $F(7)$ , и т.д.

6) после продолжительных вычислений получим  $F(26) = 4122$

7) Ответ: **4122**.

программа на языке Python:

```
def F( n ):
    if n == 1: return 1
    if n % 2 == 0:
        return n + F(n-1)
    else:
        return 2 * F(n-2)

print( F(26) )
```



# Задание 16

16

Алгоритм вычисления значения функции  $F(n)$ , где  $n$  — натуральное число, задан следующими соотношениями:

$$F(n) = 1 \text{ при } n = 1;$$

$$F(n) = 1 \text{ при } n = 2;$$

$$F(n) = 2 + F(n - 1), \text{ если } n > 2 \text{ и при этом если } n \text{ чётно};$$

$$F(n) = 3 \cdot n + F(n - 2), \text{ если } n > 2 \text{ и при этом } n \text{ нечётно.}$$

Чему равно значение функции  $F(43)$ ?

Ответ: \_\_\_\_\_.

```
def f(n):
    if n == 1 or n == 2:
        return 1
    elif n > 2 and n % 2 == 0:
        return 2 + f(n - 1)
    elif n > 2 and n % 2 == 1:
        return 3 * n + f(n - 2)
print(f(43))
```

# Задание 17

- 1) **(П. Волгин)** В файле **17-278.txt** содержится последовательность целых чисел. Элементы последовательности могут принимать целые значения от 0 до 1000. Определите сначала количество пар, в которых оба числа больше, чем сумма всех цифр «7» в восьмеричной записи всех чисел в файле, а затем минимальную из сумм таких пар. Под парой подразумевается два идущих подряд элемента последовательности.
- 2) **(В. Селезнёв)** В файле **17-280.txt** содержится последовательность целых чисел. Элементы последовательности могут принимать целые значения от 0 до 1000. В этой последовательности рассматриваются шестёрки идущих подряд чисел, в которых первые три элемента образуют возрастающую арифметическую прогрессию, а последние три элемента – возрастающую геометрическую, или наоборот – первые три элемента образуют возрастающую геометрическую прогрессию, а последние три – возрастающую арифметическую прогрессию. Выберите такие из указанных шестёрок чисел, для которых знаменатель геометрической прогрессии такой же, как и разность арифметической прогрессии. В ответе запишите два числа: сначала количество подходящих шестёрок, а затем наибольшую из их сумм.

**17-278.txt    17-280.txt**

**Ответы: 1) 71 553  
2) 24 27600**

# Задание 17

17

В файле<sup>1</sup> содержится последовательность целых чисел. Элементы последовательности могут принимать целые значения от  $-10\,000$  до  $10\,000$  включительно. Определите и запишите в ответе сначала количество пар элементов последовательности, в которых хотя бы один элемент пары превосходит  $700$ , затем максимальную из сумм квадратов элементов таких пар. В данной задаче под парой подразумевается два подряд идущих элемента последовательности. Например, для последовательности из пяти элементов:

701; 7; 9; -3; 800 — ответ

2

640009

```
a = []
with
open('C:\\Users\\ЕГЕ1\\Desktop\\EGE2022
\\17\\17var11.txt') as f:
    for i in f.readlines():
        a.append(int(i))
print(a)
k = 0
m = 0
for i in range(0, len(a)-1):
    if a[i] > 700 or a[i+1] > 700:
        k+=1
        n = a[i]**2+a[i+1]**2
        if n > m:
            m = n
print(k,m)
```

# Задание 23

Исполнитель преобразует число на экране. У исполнителя есть две команды, которым присвоены номера:

1. Прибавить 1
2. Умножить на 2

Первая команда увеличивает число на экране на 1, вторая умножает его на 2. Программа для исполнителя – это последовательность команд. Сколько существует программ, для которых при исходном числе 1 результатом является число 20, и при этом траектория вычислений содержит число 10?

**Решение (теоретическое)**

**Решение (электронные таблицы Excel, LibreOffice Calc)**

**Решение (рекурсивная программа, Python)**

1) на Паскале:

```
function numProg( start, x: integer ): integer;
var K: integer;
begin
  if x < start then numProg := 0
  else if x = start then numProg := 1
  else begin
    K := numProg( start, x-1 );
    if x mod 2 = 0 then
      K := K + numProg( start, x div 2 );
    numProg := K;
  end;
end;

begin
  writeln( numProg(1,10)*numProg(10,20) );
end.
```

# Задание 23

```
def np(start,x,block):  
    if x < start:  
        return 0  
    elif start == block:  
        return 0  
    elif x == start:  
        return 1  
    else:  
        return  
np(start+3,x,block)+np(start+4,x,block)+np(  
start*3,x,block)  
print(np(1,7,30)*np(7,30,100))
```

23

Исполнитель Вычислитель преобразует число, записанное на экране.

У исполнителя есть три команды, которым присвоены номера:

1. Прибавить 3
2. Прибавить 4
3. Умножить на 3

Первая из них увеличивает число на экране на 3, вторая увеличивает его на 4, третья умножает его на 3.

Программа для Вычислителя — это последовательность команд.

Сколько существует таких программ, которые преобразуют исходное число 1 в число 30 и при этом траектория вычислений программы содержит число 7?

Траектория вычислений программы — это последовательность результатов выполнения всех команд программы. Например, для программы 132 при исходном числе 2 траектория будет состоять из чисел 5, 15, 19.

# Задание 25

```
def DMAX(n):  
    for i in range(n//2+1,1,-1):  
        if n % i == 0:  
            return i  
    return 0  
  
def DMIN(n):  
    for i in range(2, n//2+2):  
        if n % i == 0:  
            return i  
    return 0  
  
k = 0  
x = 450001  
while k <= 5:  
    m = DMAX(x) - DMIN(x)  
    if (m != 0) and (m % 7 == 0):  
        k += 1  
        print(x, m)  
    x += 1
```

25

Напишите программу, которая перебирает целые числа, большие 450 000, в порядке возрастания и ищет среди них такие, для которых наибольший натуральный делитель, не равный самому числу, не является простым числом. Программа должна найти и вывести первые 6 таких чисел и соответствующие им значения упомянутых делителей.

Формат вывода: для каждого из 6 таких найденных чисел в отдельной строке сначала выводится само число, затем упомянутый делитель. Строки выводятся в порядке возрастания найденных чисел.

Например, для числа 105 наибольший натуральный делитель 35 не является простым, для числа 15 наибольший натуральный делитель 5 — простое число, а для числа 13 такого делителя не существует.

Ответ:

...	...

**Иванова Наталья  
Михайловна,**  

---

**учитель математики и информатики**

**215240 Смоленская область  
с. Новодугино  
ул. Чкалова, д. 27  
МКОУ «Новодугинская СШ»**

**Контакты**

**Телефон: (48138) 2-18-79**

**E-mail: [shcool\\_2002@mail.ru](mailto:shcool_2002@mail.ru)**