

**Единый государственный экзамен
по ИНФОРМАТИКЕ и ИКТ**

Инструкция по выполнению работы

Экзаменационная работа состоит из двух частей, включающих в себя 27 заданий. Часть 1 содержит 23 задания с кратким ответом, часть 2 содержит 4 задания с развёрнутым ответом.

На выполнение экзаменационной работы по информатике и ИКТ отводится 3 часа 55 минут (235 минут).

Ответы к заданиям 1–23 записываются в виде числа, последовательности букв или цифр. Ответы запишите в поле ответа в тексте работы, а затем перенесите в бланк ответов № 1.

КИМ

Ответ: 23

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

Бланк

Задания 24–27 требуют развёрнутого ответа. В бланке ответов № 2 укажите номер задания и запишите его полное решение.

Все бланки ЕГЭ заполняются яркими чёрными чернилами. Допускается использование гелевой, капиллярной или перьевой ручек.

При выполнении заданий можно пользоваться черновиком. Записи в черновике не учитываются при оценивании работы.

Баллы, полученные Вами за выполненные задания, суммируются. Постарайтесь выполнить как можно больше заданий и набрать наибольшее количество баллов.

Желаем успеха!

В экзаменационных заданиях используются следующие соглашения.

1. Обозначения для логических связей (операций):

- a) отрицание (инверсия, логическое НЕ) обозначается \neg (например, $\neg A$);
- b) конъюнкция (логическое умножение, логическое И) обозначается \wedge (например, $A \wedge B$) либо $\&$ (например, $A \& B$);
- c) дизъюнкция (логическое сложение, логическое ИЛИ) обозначается \vee (например, $A \vee B$) либо $|$ (например, $A | B$);
- d) следование (импликация) обозначается \rightarrow (например, $A \rightarrow B$);
- e) тождество обозначается \equiv (например, $A \equiv B$). Выражение $A \equiv B$ истинно тогда и только тогда, когда значения A и B совпадают (либо они оба истинны, либо они оба ложны);
- f) символ 1 используется для обозначения истины (истинного высказывания); символ 0 – для обозначения лжи (ложного высказывания).

2. Два логических выражения, содержащих переменные, называются равносильными (эквивалентными), если значения этих выражений совпадают при любых значениях переменных. Так, выражения $A \rightarrow B$ и $(\neg A) \vee B$ равносильны, а $A \vee B$ и $A \wedge B$ неравносильны (значения выражений разные, например, при $A = 1, B = 0$).

3. Приоритеты логических операций: инверсия (отрицание), конъюнкция (логическое умножение), дизъюнкция (логическое сложение), импликация (следование), тождество. Таким образом, $\neg A \wedge B \vee C \wedge D$ означает то же, что и $((\neg A) \wedge B) \vee (C \wedge D)$.

Возможна запись $A \wedge B \wedge C$ вместо $(A \wedge B) \wedge C$. То же относится и к дизъюнкции: возможна запись $A \vee B \vee C$ вместо $(A \vee B) \vee C$.

4. Обозначения Мбайт и Кбайт используются в традиционном для информатики смысле – как обозначения единиц измерения, чьё соотношение с единицей «байт» выражается степенью двойки.

Часть 1

Ответами к заданиям 1–23 являются число, последовательность букв или цифр, которые следует записать в БЛАНК ОТВЕТОВ № 1 справа от номера соответствующего задания, начиная с первой клеточки, без пробелов, запятых и других дополнительных символов. Каждый символ пишете в отдельной клеточке в соответствии с приведёнными в бланке образцами.

1 Найдите значение выражения $8F - 80$ в шестнадцатеричной системе счисления. Ответ дайте в десятичной системе счисления.

Ответ: _____.

2 Логическая функция F задаётся выражением $(x \wedge \neg y) \vee (y \equiv z) \vee \neg w$. На рисунке приведён фрагмент таблицы истинности функции F , содержащий все наборы аргументов, при которых функция F ложна. Определите, какому столбцу таблицы истинности функции F соответствует каждая из переменных w, x, y, z . Все строки в представленном фрагменте разные.

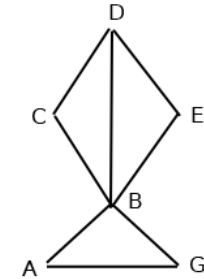
Перем.1	Перем.2	Перем.3	Перем.4
???	???	???	???
	0		
1	0		0
1		0	0

В ответе напишите буквы w, x, y, z в том порядке, в котором идут соответствующие им столбцы (без разделителей).

Ответ: _____.

3 На рисунке справа схема дорог Н-ского района изображена в виде графа, в таблице содержатся сведения о дорогах между населёнными пунктами (звездочка означает, что дорога между соответствующими городами есть).

	1	2	3	4	5	6
1		*		*		
2	*			*		*
3				*	*	
4	*	*	*		*	*
5			*	*		
6		*		*		



Так как таблицу и схему рисовали независимо друг от друга, то нумерация населённых пунктов в таблице никак не связана с буквенными обозначениями на графе. Определите номера населённых пунктов A и G в таблице. В ответе запишите числа в порядке возрастания без разделителей

Ответ: _____.

4 В фрагменте базы данных представлены сведения о родственных отношениях. На основании приведённых данных определите количество человек, у которых есть брат с разницей не более 5 лет.

ID	Фамилия ИО	Пол	Год
2053	Сухорук К.К.	М	1975
2065	Лопухова В.А.	Ж	1980
2086	Зарецкий А.А.	М	1972
2097	Сухорук Е.К.	Ж	2004
2118	Ларина О.Д.	Ж	1996
2124	Сухорук И.К.	М	2001
2135	Кольцова Т.Х.	Ж	1995
2156	Рац А.П.	М	1993
2181	Сухорук Т.Н.	М	2015
2203	Сухорук П.И.	Ж	2018
2052	Гнатюк О.А.	М	1952

ID родителя	ID ребенка
2065	2097
2053	2118
2052	2065
2052	2086
2053	2135
2052	2053
2065	2124
2086	2156
2156	2181
2156	2203

Ответ: _____.

5

Для передачи данных используется двоичный код. Сообщение содержит только буквы А, Б, В или Г, для букв А, Б и В используются следующие кодовые слова:
А – 0, Б – 101, В – 111.

Найдите кодовое слово минимальной длины для Г при котором сохраняется прямое условие Фано. Если таких кодовых слов несколько, укажите кодовое слово с минимальным двоичным значением.

Ответ: _____.

6

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: складываются все цифры двоичной записи, если
 - а) сумма нечетная к числу дописывается 11,
 - б) сумма четная, дописывается 00.

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 R, которое превышает 114 и может являться результатом работы алгоритма. В ответе это число запишите в десятичной системе счисления.

Ответ: _____.

7

Дан фрагмент электронной таблицы. Из ячейки D2 в ячейку E1 была скопирована формула. При копировании адреса ячеек в формуле автоматически изменились. Каким стало числовое значение формулы в ячейке E1?

	A	B	C	D	E
1	1	10	100	1000	
2	2	20	200	=B2+C\$3	20000
3	3	30	300	3000	30000
4	4	40	400	4000	40000

Ответ: _____.

8

Запишите число, которое будет напечатано в результате выполнения следующей программы.

Бейсик	Python
<pre>DIM S, N AS INTEGER S = 0 N = 170 WHILE S + N < 325 S = S + 25 N = N - 5 WEND PRINT S</pre>	<pre>s = 0 n = 170 while s + n < 325: s = s + 25 n = n - 5 print(s)</pre>
C++	Паскаль
<pre>#include <iostream> using namespace std; int main() { int s = 0, n = 170; while (s + n < 325) { s = s + 25; n = n - 5; } cout << s << endl; return 0; }</pre>	<pre>var s, n: integer; begin s := 0; n := 170; while s + n < 325 do begin s := s + 25; n := n - 5; end; writeln(s); end.</pre>

Ответ: _____.

9

Графический файл с разрешением 1024x600 на жестком диске занимает не более 120 КБайт. Определите максимальное количество цветов, которое может использоваться для кодирования данного изображения.

Ответ: _____.

10 Все 6-буквенные слова, составленные из букв А, О, У, записаны в обратном алфавитном порядке. Вот начало списка:

1. УУУУУУ
2. УУУУУО
3. УУУУУА
4. УУУУОУ

.....

На каком месте от начала списка находится слово ОУУУОО.

Ответ: _____.

11 Ниже на четырех языках программирования записан рекурсивный алгоритм F.

Бейсик	Python
<pre>SUB F (n) IF n > 0 THEN F(n \ 4) PRINT n F(n - 1) END IF END SUB</pre>	<pre>def F(n): if n > 0: F(n // 4) print(n) F(n - 1)</pre>
C++	Паскаль
<pre>void F(int n){ if (n > 0){ F(n / 4) std::cout <<n; F(n - 1); } }</pre>	<pre>procedure F(n: integer); begin if n > 0 then begin F(n div 4); write(n); F(n - 1); end end;</pre>

В качестве ответа укажите последовательность цифр, которая будет напечатана на экране в результате вызова F(5).

Ответ: _____.

12 Даны IP-адрес и адрес сети.

IP-адрес: 153.82.140.123

Адрес сети: 153.82.136.0

Определите третий слева октет маски подсети.

Ответ: _____.

13 При регистрации в компьютерной системе каждому пользователю выдается пароль, состоящий из 10 символов и содержащий только символы из 26-символьного латинского алфавита. В базе данных для хранения сведений о каждом пользователе отведено одинаковое и минимально возможное целое число байт. При этом используют посимвольное кодирование паролей, все символы кодируют одинаковым и минимально возможным количеством бит. Кроме собственно пароля, для каждого пользователя в системе хранятся дополнительные сведения, для чего отведено 14 байт на одного пользователя. Определите объём памяти (в байтах), необходимый для хранения сведений о 5 пользователях.

Ответ: _____.

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

1. заменить (v, w)
2. нашлось (v)

Первая команда заменяет в строке первое слева вхождение цепочки v на цепочку w , вторая проверяет, встречается ли цепочка v в строке исполнителя Редактор. Если она встречается, то команда возвращает логическое значение «истина», в противном случае возвращает значение «ложь».

Какая строка получится в результате применения приведённой ниже программы к строке, состоящей из одной единицы и 75 стоящих справа от нее нулей? В ответе запишите сколько нулей будет в конечной строке.

НАЧАЛО

ПОКА нашлось (10) ИЛИ нашлось (1)

ЕСЛИ нашлось (10)

ТО заменить (10, 001)

ИНАЧЕ заменить (1, 00)

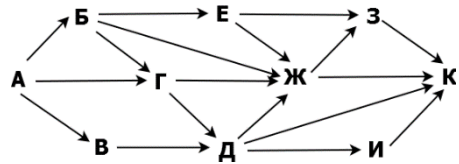
КОНЕЦ ЕСЛИ

КОНЕЦ ПОКА

КОНЕЦ

Ответ: _____.

15 На рисунке представлена схема дорог. По каждой дороге можно двигаться только в одном направлении, указанном стрелкой. Сколько существует различных путей из города А в город К, проходящих через город Г и НЕ проходящих через город З?



Ответ: _____.

16 Сколько единиц в двоичной записи числа, являющимся результатом следующего выражения?

$$4^{14} + 2^{32} - 4$$

Ответ: _____.

17 В таблице приведены запросы и количество найденных по ним страниц некоторого сегмента сети Интернет.

Запрос	Количество страниц(тыс.)
Вебинар	50
Информатика	80
Ум	100
Вебинар Информатика	110
Информатика & Ум	30
Вебинар & Информатика & Ум	5
Вебинар Информатика Ум	165

Какое количество страниц (в тысячах) будет найдено по запросу Вебинар & Ум

Ответ: _____.

18 Укажите наименьшее значение А, при котором выражение

$$(y+3x < A) \vee (x > 20) \vee (y > 40)$$

истинно для любых целых положительных значений x и y .

Ответ: _____.

19 Ниже представлен фрагмент программы, обрабатывающей одномерный целочисленный массив с индексами от 0 до 10. Значения элементов равны 5; 1; 6; 7; 8; 8; 7; 7; 6; 9; 5 соответственно, т.е. $A[0]=5$; $A[1]=1$ и т.д. Определите значение переменной s после выполнения следующего фрагмента программы

C++	Python
<pre>s = 0; for (i=0; i<10; i++) s = s + A[i+1] - A[i];</pre>	<pre>s = 0 for i in range(10): s = s + A[i+1] - A[i]</pre>
Паскаль	Бэйсик
<pre>s := 0; for i:=0 to 9 do begin s := s + A[i+1] - A[i] end;</pre>	<pre>S = 0 FOR i = 0 TO 9 S = S + A(i+1) - A(i); NEXT i</pre>

Ответ: _____.

20 Укажите наибольшее десятичное число, при вводе которого на экране сначала напечатается 3, а затем 6.

Бейсик	Python
<pre>DIM X, L, M AS INTEGER INPUT X L = 0 M = 0 WHILE X > 0 L = L + 1 IF X MOD 2 <> 0 THEN M = M + (X MOD 8) ENDIF X = X \ 8 WEND PRINT L PRINT M</pre>	<pre>x = int(input()) L = 0 M = 0 while x > 0 : L = L+1 if (x % 2) != 0: M = M + x % 8 x = x // 8 print(L) print(M)</pre>
C++	Паскаль
<pre>#include <iostream> using namespace std; int main(void) { int L, M, x; cin >> x; L = 0; M = 0; while (x > 0) { L = L+1; if (x % 2 != 0) { M = M + x % 8; } x = x / 8; } cout << L << " " << M; }</pre>	<pre>var x, L, M: integer; begin readln(x); L:=0; M:=0; while x > 0 do begin L:=L + 1; if (x mod 2) <> 0 then M:= M + x mod 8; x:= x div 8; end; writeln(L); write(M); end.</pre>

Ответ: _____.

21 Какое значение будет выведено на экран после выполнения данной программы?

Бейсик	Python
<pre> DIM A, B, T, M, R AS INTEGER A = -20: B = 20 M = A: R = F(A) FOR T = A TO B IF F(T) > R THEN M = T R = F(T) END IF NEXT T PRINT M+24 FUNCTION F(x) F := 2*(x-19)*(x-19)+7 END FUNCTION </pre>	<pre> def f(x): return 2*(x-19)*(x-19)+7 a = -20 b = 20 M = a R = f(a) for t in range(a, b+1): if (f(t) > R): M = t R = f(t) print(M+24) </pre>
C++	Паскаль
<pre> #include <iostream> using namespace std; int F(int x) { return 2*(x-19)*(x-19)+7 } int main() { int a, b, t, M, R; a = -20; b = 20; M = a; R = F(a); for (t=a; t<=b; t++){ if (F(t) > R) { M = t; R = F(t); } } cout << (M+24) << endl; } </pre>	<pre> var a,b,t,M,R :integer; Function F(x: integer):integer; begin F := 2*(x-19)*(x-19)+7; end; BEGIN a := -20; b := 20; M := a; R := F(a); for t := a to b do begin if (F(t) > R) then begin M := t; R := F(t); end; end; write(M+24); END. </pre>

Ответ: _____.

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

1. Прибавить 1
2. Умножить на 2

Программа для исполнителя Калькулятор – это последовательность команд. Сколько существует программ, для которых при исходном числе 1 результатом является число 21, при этом траектория вычислений содержит число 10 и не содержит число 17?

Ответ: _____.

23 Сколько существует различных наборов значений логических переменных x_1, x_2, \dots, x_8 , которые удовлетворяют всем перечисленным ниже условиям?

- $$(x_1 \vee \neg x_2) \rightarrow (x_3 \vee \neg x_4) = 1$$
- $$(x_3 \vee \neg x_4) \rightarrow (x_5 \vee \neg x_6) = 1$$
- $$(x_5 \vee \neg x_6) \rightarrow (x_7 \vee \neg x_8) = 1$$

Ответ: _____.

Не забудьте перенести все ответы в бланк ответов № 1 в соответствии с инструкцией по выполнению работы.

Часть 2

Для записи ответов на задания этой части (24–27) используйте **БЛАНК ОТВЕТОВ № 2**. Запишите сначала номер задания (24, 25 и т. д.), а затем полное решение. Ответы записывайте чётко и разборчиво.

24

На вход программы поступают 4 неотрицательных целых числа, не превышающие 1000, среди которых могут быть одинаковые. Нужно написать программу, которая выводит количество чисел, не кратных 3, и максимальное из этих чисел. Если среди входных данных нет чисел, не кратных трём, программа должна вывести слово «NO». Программист написал программу неправильно.

Последовательно выполните следующее.

1. Напишите, что выведет эта программа при вводе чисел 7 15 8 21.
2. Приведите пример такой последовательности, содержащей число, кратное 3, при вводе которой программа выведет правильный ответ.
3. Найдите все ошибки в этой программе (их может быть одна или несколько). Известно, что каждая ошибка затрагивает только одну строку и может быть исправлена без изменения других строк. Для каждой ошибки:
 - 1) выпишите строку, в которой сделана ошибка;
 - 2) укажите, как исправить ошибку, т.е. приведите правильный вариант строки.

Бейсик	Python
<pre> COUNT = 0 MAXIMUM = 1000 FOR I = 1 TO 4 INPUT X IF X MOD 3 <> 0 THEN COUNT = COUNT + 1 IF X > MAXIMUM THEN MAXIMUM = I END IF END IF NEXT I IF COUNT > THEN PRINT COUNT PRINT MAXIMUM ELSE PRINT "NO" END IF </pre>	<pre> count = 0 maximum = 1000 for i in range(4): x = int(input()) if x % 3 != 0: count = count + 1 if x > maximum: maximum = i if count > 0: print(count) print(maximum) else: print("NO") </pre>
C++	Паскаль
<pre> #include <iostream> int main(){ int i, x, maximum; count = 0; maximum = 1000; for (i=1; i<= 4; i++) { std::cin >> x; if (x % 3 != 0) { count = count + 1; if (x > maximum) maximum = i; } } if (count > 0) { std::cout << count <<endl; std::cout << maximum } else std::cout << "NO"; return 0; } </pre>	<pre> var i, x: integer; var maximum, count: integer; begin count := 0; maximum := 1000; for i:=1 to 4 do begin read(x); if x mod 3 <> 0 then begin count := count + 1; if x > maximum then maximum := i; end; end; if count > 0 then begin writeln(count); writeln(maximum); end else writeln('NO'); end. </pre>

25 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, который находит количество элементов массива, меньших 100, не делящихся на 3, после чего заменяет в массиве соответствующие значения на найденное количество. После чего выводит полученный массив на экран.

Бейсик	Python
<pre>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</pre>	<pre># допускается также # использовать две # целочисленные # переменные j, k a = [] n = 40 for i in range(n): a.append(int(input())) ...</pre>
C++	Паскаль
<pre>#include <iostream> #define n 40 int main() { int a[n]; int i, j, k; for (i = 0; i < n; i++) std::cin >> a[i]; ... return 0; }</pre>	<pre>const n = 40; var a: array [0..n-1] of integer; i, j, k: integer; begin for i := 0 to n-1 do readln(a[i]); ... end.</pre>

26 Два игрока, Паша и Вася, играют в следующую игру. Перед игроками лежит куча камней. Игроки ходят по очереди, первый ход делает Паша. За один ход игрок может добавить в кучу **один или четыре камня** или увеличить количество камней в куче **в пять раз**. Игра завершается в тот момент, когда количество камней в куче становится не менее **69**. Победителем считается игрок, сделавший последний ход, то есть первым получивший кучу, в которой будет 69 или больше камней. В начальный момент в куче было S камней, $1 \leq S \leq 68$.

Задание 1. а) Укажите все такие значения числа S, при которых Паша может выиграть в один ход. Обоснуйте, что найдены все нужные значения S, и укажите выигрывающий ход для каждого указанного значения S. б) Укажите такое значение S, при котором Паша не может выиграть за один ход, но при любом ходе Паши Вася может выиграть своим первым ходом. Опишите выигрышную стратегию Васи.

Задание 2. Укажите 2 таких значения S, при которых у Паши есть выигрышная стратегия, причём Паша не может выиграть за один ход и может выиграть своим вторым ходом независимо от того, как будет ходить Вася. Для каждого указанного значения S опишите выигрышную стратегию Паши.

Задание 3. Укажите хотя бы одно значение S, при котором у Васи есть выигрышная стратегия, позволяющая ему выиграть первым или вторым ходом при любой игре Паши, и у Васи нет стратегии, которая позволит ему гарантированно выиграть первым ходом. Для указанного значения S опишите выигрышную стратегию Васи. Постройте дерево всех партий, возможных при этой выигрышной стратегии Васи (в виде рисунка или таблицы).

27

На вход программы поступает последовательность из N целых положительных чисел, все числа в последовательности различны. Рассматриваются все пары различных элементов последовательности (элементы пары не обязаны стоять в последовательности рядом, порядок элементов в паре не важен). Необходимо определить количество пар, для которых произведение элементов не делится на 34.

Описание входных и выходных данных

В первой строке входных данных задаётся количество чисел N ($1 \leq N \leq 1000$). В каждой из последующих N строк записано одно целое положительное число, не превышающее 10 000. В качестве результата программа должна напечатать одно число: количество пар, в которых произведение элементов не кратно 34.

Пример входных данных:

5
3
4
10
11
17

Пример выходных данных для приведённого выше примера входных данных:

8

Пояснение. Из четырёх заданных чисел можно составить 10 попарных произведений: $3 \cdot 4$, $3 \cdot 10$, $3 \cdot 11$, $3 \cdot 17$, $4 \cdot 10$, $4 \cdot 11$, $4 \cdot 17$, $10 \cdot 11$, $10 \cdot 17$, $11 \cdot 17$ (результаты: 12, 30, 33, 51, 40, 44, 68, 110, 170, 187). Из них на 34 не делятся 8 произведения ($3 \cdot 4=12$, $3 \cdot 10=30$, $3 \cdot 11=33$, $3 \cdot 17=51$, $4 \cdot 10=40$, $4 \cdot 11=44$, $10 \cdot 11=110$, $11 \cdot 17=187$).

Требуется написать эффективную по времени и по памяти программу для решения описанной задачи.

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

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

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

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

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

Вы можете сдать одну программу или две программы решения задачи (например, одна из программ может быть менее эффективна). Если Вы сдадите две программы, то каждая из них будет оцениваться независимо от другой, итоговой станет большая из двух оценок.

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

Система оценивания экзаменационной работы по информатике и ИКТ**Часть 1**

За правильный ответ на задания 1–23 ставится 1 балл; за неверный ответ или его отсутствие – 0 баллов.

№ задания	Ответ
1	15
2	wzyx
3	35
4	5
5	100
6	115
7	3010
8	200
9	2
10	248
11	1514321
12	248
13	105
14	152
15	8
16	27
17	20
18	101
19	0
20	425
21	4
22	14
23	121

**Спасибо всем, кто принимал участие в
поиске и исправлении ошибок!
Вместе мы делаем подготовку к экзамену
лучше!**

Часть 2

24

На вход программы поступают 4 неотрицательных целых числа, не превышающие 1000, среди которых могут быть одинаковые. Нужно написать программу, которая выводит количество чисел, не кратных 3, и максимальное из этих чисел. Если среди входных данных нет чисел, не кратных трём, программа должна вывести слово «NO». Программист написал программу неправильно.

Бейсик	Python
<pre> COUNT = 0 MAXIMUM = 1000 FOR I = 1 TO 4 INPUT X IF X MOD 3 <> 0 THEN COUNT = COUNT + 1 IF X > MAXIMUM THEN MAXIMUM = X END IF END IF NEXT I IF COUNT > 0 THEN PRINT COUNT PRINT MAXIMUM ELSE PRINT "NO" END IF </pre>	<pre> count = 0; maximum = 1000 for i in range(4): x = int(input()) if x % 3 != 0: count = count + 1 if x > maximum: maximum = x if count > 0: print(count) print(maximum) else: print("NO") </pre>
C++	Паскаль
<pre> #include <iostream> int main(){ int i, x, maximum; count = 0; maximum = 1000; for (i=1; i<= 4; i++) { std::cin >> x; if (x % 3 != 0) { count = count + 1; if (x > maximum) maximum = x; } } if (count > 0) { std::cout << count <<endl; std::cout << maximum else std::cout << "NO"; return 0; } </pre>	<pre> var i, x: integer; var maximum, count: integer; begin count := 0; maximum := 1000; for i:=1 to 4 do begin read(x); if x mod 3 <> 0 then begin count := count + 1; if x > maximum then maximum := i; end; end; if count > 0 then begin writeln(count); writeln(maximum); end else writeln('NO'); end. </pre>

Последовательно выполните следующее.

1. Напишите, что выведет эта программа при вводе чисел 7 15 8 21.
2. Приведите пример такой последовательности, содержащей число, кратное 3, при вводе которой программа выведет правильный ответ.
3. Найдите все ошибки в этой программе (их может быть одна или несколько). Известно, что каждая ошибка затрагивает только одну строку и может быть исправлена без изменения других строк. Для каждой ошибки:
 - 1) выпишите строку, в которой сделана ошибка;
 - 2) укажите, как исправить ошибку (приведите правильный вариант строки).

Содержание верного ответа и указания по оцениванию (допускаются иные формулировки ответа, не искажающие его смысла)
<p>1. При вводе числа 7 15 8 21 программа сначала выведет 2, затем 1000.</p> <p>2. Примеры чисел, при вводе которых программа выводит корректный ответ: 2 3 4 1000.</p> <p>Комментарий для экспертов. Программа выводит сначала количество чисел, не кратных 3, за тем выводит 1000. Либо корректно работает при отсутствии в ряду таких чисел.</p> <p>Пример исправления для языка Паскаль:</p> <p><i>Первая ошибка:</i> maximum := 1000; Исправленная строка: maximum := 0;</p> <p><i>Вторая ошибка:</i> maximum := i; Исправленная строка: maximum := x;</p> <p><i>Пояснение для эксперта</i> В неправильной версии алгоритм никогда не зайдет в условие изменения максимума, т.к. в переменной maximum уже записано максимальное значение. Если же исправить инициализацию переменной maximum (ошибка 1), то в данной переменной будет храниться индекс считанного числа, что неверно. Поэтому необходимо в качестве нового значения переменной maximum указать считанное значение x (ошибка 2). В программах на других языках ошибочные строки и их исправления аналогичны. Незначительной опiskой, не влияющей на оценку, следует считать отсутствие служебных слов и знаков после содержательной части исправления</p>

Указания по оцениванию	Баллы
<p>В задаче требуется выполнить три действия.</p> <p>1. Указать результат программы при данном вводе. Это действие считается выполненным, если указан верный результат работы программы при заданных входных данных. Экзаменуемый не обязан объяснять, как получен этот результат, достаточно указать верное число.</p> <p>2. Указать пример ввода, при котором программа выводит верный ответ. Это действие считается выполненным, если указан пример числа, при вводе которого выводится верное сообщение (верные значения), причем результатом работы программы не может быть строка «NO». Ученик не обязан указывать, что будет выведено, и объяснять, как работает программа.</p> <p>3. Найти и исправить ошибки в программе. Это действие считается выполненным, если верно указаны обе ошибки и предложены верные варианты исправления, при этом никакие верные строки программы не указаны в качестве неверных. В исправленной строке допускаются незначительные синтаксические ошибки (лишние или пропущенные знаки препинания, неточные написания служебных слов языка). Ошибка считается исправленной, если выполнены оба следующих условия:</p> <p>а) правильно указана строка с ошибкой; б) указан такой новый вариант строки, что при исправлении другой ошибки получается правильная программа</p>	
Выполнены все три необходимых действия, и ни одна верная строка не указана в качестве ошибочной	3
<p>Не выполнены условия, позволяющие поставить 3 балла. Имеет место одна из следующих ситуаций.</p> <p>1. Выполнены два первых действия, найдена и исправлена одна ошибка в программе, ни одна верная строка не названа ошибочной.</p> <p>2. Выполнены два первых действия, найдены и исправлены две ошибки в программе, одна верная строка названа ошибочной.</p> <p>3. Выполнено одно из первых двух действий, найдены и исправлены две ошибки в программе, ни одна верная строка не названа ошибочной</p>	2
<p>Не выполнены условия, позволяющие поставить 2 или 3 балла. При этом имеет место один из следующих случаев.</p> <p>1. Выполнены два первых действия. При этом несущественно,</p>	1

<p>насколько правильно выполнено третье действие.</p> <p>2. Найдены и исправлены две ошибки в программе, не более чем одна верная строка названа ошибочной. При этом несущественно, насколько правильно выполнены действия 1 и 2.</p> <p>3. Выполнено одно из двух первых действий. Исправлена одна из двух ошибок. Не более чем одна верная строка названа ошибочной</p>	
Не выполнены условия, позволяющие поставить 1, 2 или 3 балла	0
<i>Максимальный балл</i>	3

25 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, который находит количество элементов массива, меньших 100, не делящихся на 3, после чего заменяет в массиве соответствующие значения на найденное количество. После чего выводит полученный массив на экран.

Бейсик	Python
<pre>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</pre>	<pre># допускается также # использовать две # целочисленные # переменные j, k a = [] n = 40 for i in range(n): a.append(int(input())) ...</pre>
C++	Паскаль
<pre>#include <iostream> #define n 40 int main() { int a[n]; int i, j, k; for (i = 0; i < n; i++) std::cin >> a[i]; ... return 0; }</pre>	<pre>const n = 40; var a: array [0..n-1] of integer; i, j, k: integer; begin for i := 0 to n-1 do readln(a[i]); ... end.</pre>
Содержание верного ответа и указания по оцениванию	

(допускаются иные формулировки ответа, не искажающие его смысла)	
На языке Паскаль	
<pre> k := 0; for i:=0 to n-1 do if (a[i]<100)AND(a[i] mod 3 <>0) then k:=k+1; for i:=0 to n-1 do if (a[i]<100)AND(a[i] mod 3 <>0) then a[i] := k; for i:=0 to n-1 do writeln(a[i]); </pre>	
Указания по оцениванию	Баллы
<p><i>Общие указания</i></p> <p>1. В алгоритме, записанном на языке программирования, допускается наличие отдельных синтаксических ошибок, не искажающих замысла автора программы.</p> <p>2. Эффективность алгоритма не имеет значения и не оценивается.</p> <p>3. Допускается запись алгоритма на языке программирования, отличном от языков, перечисленных в условии. В этом случае должны использоваться переменные, аналогичные описанным в условии. Если язык программирования использует типизированные переменные, описания переменных должны быть аналогичны описаниям переменных на естественном языке. Использование нетипизированных или необъявленных переменных возможно только в случае, если это допускается языком программирования; при этом количество переменных и их идентификаторы должны соответствовать условию задачи</p>	
Предложен правильный алгоритм, выдающий в качестве результата верное значение	2
<p>Не выполнены условия, позволяющие поставить 2 балла. Предложено в целом верное решение, содержащее не более одной ошибки из числа следующих:</p> <ol style="list-style-type: none"> 1) в цикле происходит выход за границу массива; 2) не инициализируется или неверно инициализируется счётчик количества искомых элементов; 3) счётчик количества искомых элементов в цикле не изменяется или изменяется неверно; 4) неверно определено условие нахождения описанных значений; 5) при проверке выполнения условия используются 	1

<p>неверные индексы;</p> <p>6) неверно осуществляется переопределение значений элементов массива;</p> <p>7) используется переменная, не объявленная в разделе описания переменных;</p> <p>8) неверно расставлены операторные скобки</p>	
Не выполнены условия, позволяющие поставить 1 или 2 балла	0
<i>Максимальный балл</i>	2

26 Два игрока, Паша и Вася, играют в следующую игру. Перед игроками лежит куча камней. Игроки ходят по очереди, первый ход делает Паша. За один ход игрок может добавить в кучу **один или четыре камня** или увеличить количество камней в куче **в пять раз**. Игра завершается в тот момент, когда количество камней в куче становится не менее **69**. Победителем считается игрок, сделавший последний ход, то есть первым получивший кучу, в которой будет 69 или больше камней. В начальный момент в куче было S камней, $1 \leq S \leq 68$.

Задание 1.

а) Укажите все такие значения числа S , при которых Паша может выиграть в один ход. Обоснуйте, что найдены все нужные значения S , и укажите выигрывающий ход для каждого указанного значения S .

б) Укажите такое значение S , при котором Паша не может выиграть за один ход, но при любом ходе Паши Вася может выиграть своим первым ходом. опишите выигрышную стратегию Васи.

Задание 2.

Укажите 2 таких значения S , при которых у Паши есть выигрышная стратегия, причём Паша не может выиграть за один ход и может выиграть своим вторым ходом независимо от того, как будет ходить Вася. Для каждого указанного значения S опишите выигрышную стратегию Паши.

Задание 3.

Укажите хотя бы одно значение S , при котором у Васи есть выигрышная стратегия, позволяющая ему выиграть первым или вторым ходом при любой игре Паши, и у Васи нет стратегии, которая позволит ему гарантированно выиграть первым ходом. Для указанного значения S опишите выигрышную стратегию Васи. Постройте дерево всех партий, возможных при этой выигрышной стратегии Васи (в виде рисунка или таблицы).

Содержание верного ответа и указания по оцениванию (допускаются иные формулировки ответа, не искажающие его смысла)								
1. а) ≥ 14 . При количестве камней в куче от 14 Паше необходимо увеличить количество камней в пять раз, тем самым получив 70 или более камней. б) 13. Паша своим первым ходом может сделать 14, 17 или 65 камней, после этого Вася увеличивает количество камней в пять раз, получая 70, 85 или 325 камней в куче.								
2. 9, 12. Для данных случаев Паше необходимо прибавить 4 камня к куче из 9 камней, либо 1 камень к куче из 12 камней, и получить кучу из 13 камней. После чего игра сводится к стратегии, описанной в пункте 1б.								
	Паша		Вася		Паша			
9	+4	13	+1	14	*5	70		
12	+1		+4	17	*5	85		
			*5	65	*5	325		
3. 8. Своим первым ходом Паша может сделать количество камней в куче 9, 12 или 40. Если Паша делает ход «увеличить в пять раз», тогда Вася выигрывает своим первым ходом, увеличивая количество камней в пять раз. Для случая 9 и 12 камней Вася использует стратегию, указанную в п.2. Данную стратегию можно представить в виде таблицы.								
	Паша		Вася		Паша		Вася	
8	+1	9	+4	13	+1	14	*5	70
	+4	12	+1		+4	17	*5	85
	*5	40	*5		*5	65	*5	325
ИЛИ 11. Своим первым ходом Паша может сделать 12, 15 или 55 камней. Если количество камней в куче 15 или 55 Васе необходимо увеличить количество камней в 5 раз. Для случая 12 камней Вася использует стратегию, указанную в п.2. Представим стратегию в виде таблицы.								
	Паша		Вася		Паша		Вася	
11	+1	12	+1	13	+1	14	*5	70
					+4	17	*5	85
					*5	65	*5	325
	+4	15	*5	75				
*5	55	275						

Указания по оцениванию	Баллы
<p><i>Предварительные замечания</i> В задаче от ученика требуется выполнить три задания. Их трудность возрастает. Количество баллов в целом соответствует количеству выполненных заданий. Пункт 1а считается выполненным, если указаны все варианты. Пункт 1б считается выполненным, если (i) правильно найдено описанное в условии значение. Первое задание считается выполненным полностью, если выполнены полностью оба пункта: 1а и 1б. Замечание для проверяющего. Описать стратегию игрока – значит описать, какой ход он должен сделать в любой ситуации, которая ему может встретиться при различной игре противника (см. условие задачи). Есть два основных способа сделать это. (1) Можно построить дерево всех партий, возможных при выбранной стратегии, и убедиться, что все заключительные позиции являются выигрышными для игрока, реализующего стратегию. (2) Можно свести задачу к рассмотренным выше позициям. Например, выигрышную стратегию для игрока, который ходит первым, можно описать, указав ход, ведущий в позицию, для которой известна выигрышная стратегия для игрока, который ходит вторым. Чтобы подобным образом описать выигрышную стратегию для игрока, который ходит вторым, нужно перебрать все возможные первые ходы первого игрока и убедиться, что для всех полученных позиций мы знаем выигрышную стратегию для игрока, который ходит первым. Экзаменуемый может описывать стратегию любым удобным ему способом. Существенно (повторим), чтобы (1) для каждой позиции, которая может встретиться игроку, реализующему стратегию, было понятно, какой ход он должен сделать, и (2) было показано, что все возможные заключительные позиции выигрышные для этого игрока. Задание 2 считается выполненным, если (i) правильно указаны оба значения, и (ii) описаны все выигрышные стратегии. Задание 3 считается выполненным, если (i) правильно указано значение, при котором второй игрок имеет выигрышную стратегию; (ii) правильно описано дерево всех партий, возможных при этой выигрышной стратегии (в виде рисунка или таблицы). При этом допускаются арифметические ошибки, не искажающие сути решения. Во всех случаях стратегии могут быть описаны так, как это</p>	

<i>сделано в примере решения, или другим способом</i>	
Выполнены второе и третье задания. Для первого задания правильно указаны ответы на пункты 1а и 1б.	3
Не выполнены условия, позволяющие поставить 3 балла, и выполнено одно из следующих условий. <ul style="list-style-type: none"> • Выполнено третье задание. • Выполнены первое и второе задания. 	2
Не выполнены условия, позволяющие поставить 3 или 2 балла, и выполнено одно из следующих условий. <ul style="list-style-type: none"> • Первое задание выполнено, возможно, с недочётами, указанными в критериях на 3 балла. • Второе задание выполнено, возможно, с недочётами, указанными в критериях на 2 балла. • Для второго и третьего заданий во всех случаях правильно указан игрок, имеющий выигрышную стратегию 	1
Не выполнено ни одно из условий, позволяющих поставить 1, 2 или 3 балла	0
<i>Максимальный балл</i>	3

27

На вход программы поступает последовательность из N целых положительных чисел, все числа в последовательности различны. Рассматриваются все пары различных элементов последовательности (элементы пары не обязаны стоять в последовательности рядом, порядок элементов в паре не важен). Необходимо определить количество пар, для которых произведение элементов не делится на 34.

Описание входных и выходных данных

В первой строке входных данных задаётся количество чисел N ($1 \leq N \leq 1000$). В каждой из последующих N строк записано одно целое положительное число, не превышающее 10 000. В качестве результата программа должна напечатать одно число: количество пар, в которых произведение элементов не кратно 34.

Пример входных данных:

5
3
4
10
11
17

Пример выходных данных для приведённого выше примера входных данных:

8

Пояснение. Из четырёх заданных чисел можно составить 10 попарных произведений: $3 \cdot 4$, $3 \cdot 10$, $3 \cdot 11$, $3 \cdot 17$, $4 \cdot 10$, $4 \cdot 11$, $4 \cdot 17$, $10 \cdot 11$, $10 \cdot 17$, $11 \cdot 17$ (результаты: 12, 30, 33, 51, 40, 44, 68, 110, 170, 187). Из них на 34 не делятся 8 произведения ($3 \cdot 4 = 12$, $3 \cdot 10 = 30$, $3 \cdot 11 = 33$, $3 \cdot 17 = 51$, $4 \cdot 10 = 40$, $4 \cdot 11 = 44$, $10 \cdot 11 = 110$, $11 \cdot 17 = 187$).

Требуется написать эффективную по времени и по памяти программу для решения описанной задачи.

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

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

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

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

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

Вы можете сдать одну программу или две программы решения задачи (например, одна из программ может быть менее эффективна). Если Вы сдадите две программы, то каждая из них будет оцениваться независимо от другой, итоговой станет бóльшая из двух оценок.

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

Содержание верного ответа и указания по оцениванию

(допускаются иные формулировки ответа, не искажающие его смысла)

Решение эффективное по памяти и времени (Задача Б). Прямой вариант.

Решение основано на подсчете количества чисел кратных 2 и не кратных 17, кратных 17 и не кратных 2, не кратных 2 и не кратных 17. После чего, найдя количество комбинаций чисел из этих групп, получается ответ.

Обозначим за k_2 – количество чисел кратных 2 и не кратных 17, за k_{17} – количество чисел кратных 17 и не кратных 2, за k – количество чисел не кратных 2 и не кратных 17. Тогда количество чисел будет равно $k_2*(k_2-1)/2 + k_{17}*(k_{17}-1)/2 + k*(k-1)/2 + k_2*k + k_{17}*k$

Программа на языке Pascal.

```
var k2, k17, k: integer;
    n, i, x: integer;

begin
  k := 0; k2 := 0; k17 := 0;
  readln(n);
  for i:=1 to n do begin
    readln(x);
    if x mod 34 <> 0 then begin
      if x mod 2 = 0 then k2 := k2 + 1
      else if x mod 17 = 0 then k17 := k17 + 1
      else k := k + 1;
    end;
  end;

  writeln((k2*(k2-1)+k17*(k17-1)+k*(k-1))/2+k2*k+k17*k);
end.
```

Решение эффективное по памяти и времени (Задача Б). Решение от обратного.

Решение основано на подсчете количества чисел кратных 2 и не кратных 17, кратных 17 и не кратных 2, кратных 34. После чего, найдя количество комбинаций чисел из этих групп, получается ответ.

Обозначим за k_2 – количество чисел кратных 2 и не кратных 17, за k_{17} – количество чисел кратных 17 и не кратных 2, за k – количество чисел кратных 34, n – количество введенных чисел.

Тогда количество пар будет равно общему количеству пар за вычетом произведения количества чисел кратных 34 на общее количество и произведения количества чисел, кратных 2 на количества чисел кратных 17. $n*(n-1)/2 - k_{34}*(n-k_{34}) - k_{34}*(k_{34}-1)/2 - k_2*k_{17}$

Программа на языке Pascal.

```
var k2, k17, k: integer;
    n, i, x: integer;
begin
  k := 0; k2 := 0; k17 := 0;
  readln(n);
  for i:=1 to n do begin
    readln(x);
    if x mod 34 = 0 then k := k + 1
    else if x mod 2 = 0 then k2 := k2 + 1
    else if x mod 17 = 0 then k17 := k17 + 1;
  end;
  writeln(n*(n-1)/2-k34*(n-k34)-k34*(k34-1)/2-k2*k17);
end.
```

Решение не эффективное по памяти и по времени (Задача А).

```
var N: integer;
    a: array[1..10000] of integer;
    i, j, k: integer;
begin
  readln(N);
  for i:=1 to N do read(a[i]);
  k:= 0;
  for i:= 1 to N-1 do
    for j:= i+1 to N do
      if a[i]*a[j] mod 34 <> 0 then k := k + 1;
  writeln(k)
end.
```

Указания по оцениванию	Баллы
<p>Предварительные замечания.</p> <p>1. В задаче есть два задания (А и Б). Соответственно, ученик может представить две программы. В каждой из программ должно быть указано, решением какого из заданий она является. Если в работе представлена одна программа, то в ней также должно быть указано, решением какого из заданий она является.</p> <p>2. Если ученик не указал, к какому заданию относится программа, или можно предположить, что ученик ошибся в идентификации программ, необходимо следовать приведённым ниже инструкциям.</p> <p>Случай 2.1. Ученик представил только одну программу. Следует рассматривать программу как решение задания Б и оценивать её по соответствующим критериям.</p> <p>Случай 2.2. Ученик представил две программы, но указание задания есть только для одной из программ. Следует рассматривать вторую программу как ответ на оставшееся задание.</p> <p>Случай 2.3. Ученик представил две программы; ни для одной из них задание не указано, или в обоих решениях указано одно и то же задание. Следует первую (по порядку в представленных учеником материалах) программу рассматривать как ответ на задание А, а вторую – как ответ на задание Б.</p> <p>Случай 2.4. Ученик представил более двух программ. Следует рассматривать только две последние программы и соотносить их с заданиями по правилам 2.1–2.3.</p> <p>Случай 2.5. Решение, представленное в качестве решения задания А, по критериям для задания Б может быть оценено в 3 или 4 балла. При этом решение, представленное в качестве решения задания Б, оценено меньшим баллом. Следует считать, что ученик перепутал обозначения заданий и оценивать решение, представленное как решение задания А, по критериям задания Б.</p> <p><i>НАПОМИНАЕМ! Итоговый балл за задачу – это больший из баллов, полученных учеником за каждое из двух представленных решений.</i></p> <p><i>Пояснения для проверяющих.</i></p> <p>1. Задание Б является усложнением задания А. Если в качестве решения задания Б представлено решение задания А, то считается, что учеником допущена опечатка, и решение оценивается по критериям для задания А.</p>	

<p>В качестве решения задания А может быть представлена программа, которая решает задачу при произвольном количестве входных данных (как в задании Б), осуществляя полный перебор всех вариантов. Такая программа неэффективна по времени, поэтому она должна оцениваться 2 баллами.</p> <p>2. Два задания (и, соответственно, возможность для экзаменуемого представить две программы) дают ученику возможность (при его желании) сначала написать менее сложное и менее эффективное решение (задание А), которое даёт ему право получить 2 балла, а затем приступить к поиску более эффективного решения.</p> <p>3. Приведённые в п. 2.1–2.5 правила имеют целью избежать снижения баллов из-за того, что ученик перепутал обозначения заданий.</p> <p><i>Общие принципы оценивания решений</i></p> <p>4 балла ставится за эффективную и правильно работающую программу, решающую задачу в общем случае (задача Б). При этом программа может содержать до трёх синтаксических ошибок («описок»).</p> <p>3 балла ставится в случае, когда фактически задача решена и решена эффективно по времени, возможно, с хранением всех входных данных в массиве, но количество «описок» более трёх (но не более пяти) и допущено не более одной содержательной ошибки, не позволяющей усомниться в том, что экзаменуемый правильно придумал алгоритм.</p> <p>2 балла ставится, если программа, решающая задачу Б, в дополнение к неточностям, которые перечислены выше, работает неэффективно по времени и/или допущено до трёх упомянутых выше содержательных ошибок. Количество допустимых «описок» – до семи.</p> <p>2 балла также ставится за правильное решение упрощенной задачи (задача А).</p> <p>1 балл ставится, если программа написана неверно, но из описания алгоритма и общей структуры программы видно, что экзаменуемый в целом правильно представляет путь решения задачи.</p> <p>Далее эти общие принципы уточнены</p>	
<p>Критерии оценивания задания А</p>	
<p>При решении задачи А программа верно находит требуемую сумму для любых 6 пар исходных данных. Допускается до пяти синтаксических и приравненных к ним ошибок (см. критерии оценивания задания Б на 4 балла)</p>	2
<p>Не выполнены условия, позволяющие поставить 2 балла. Из</p>	1

описания алгоритма и общей структуры программы видно, что Экзаменуемый в целом правильно представляет путь решения задачи. Допускается любое количество «описок»	
Не выполнены критерии, позволяющие поставить 1 или 2 балла	0
<i>Максимальный балл для задания А</i>	2
Критерии оценивания задания Б	
Программа правильно работает для любых соответствующих условию входных данных и при этом эффективна как по времени, так и по памяти, т.е. не используются массивы и другие структуры данных (в том числе стек рекурсивных вызовов), размер которых зависит от количества входных элементов, а время работы пропорционально этому количеству. Возможно использование массивов и динамических структур данных при условии, что в них в каждый момент времени хранится фиксированное количество элементов, требующих для хранения меньше 1Кб. Программа может содержать не более трёх синтаксических ошибок следующих видов: 1) пропущен или неверно указан знак пунктуации; 2) неверно написано или пропущено зарезервированное слово языка программирования; 3) не описана или неверно описана переменная; 4) применяется операция, недопустимая для соответствующего типа данных. К синтаксическим ошибкам приравнивается использование неверного типа данных. Если одна и та же ошибка встречается несколько раз, она считается за одну ошибку	4
Не выполнены условия, позволяющие поставить 4 балла. Программа в целом работает правильно для любых входных данных произвольного размера. Время работы пропорционально количеству введённых чисел; правильно указано, какие величины должны вычисляться по ходу чтения элементов последовательности чисел. Количество синтаксических ошибок («описок») указанных выше видов – не более пяти. Используемая память, возможно, зависит от количества прочитанных чисел (например, входные данные запоминаются в массиве, контейнере STL в C++ или другой структуре данных). Допускается ошибка при вводе и выводе данных, не влияющая на содержание решения. Программа может содержать не более пяти синтаксических и приравненных к ним ошибок, описанных в критериях на 4 балла.	3

Кроме того, допускается наличие одной ошибки, принадлежащей к одному из следующих видов: 1) ошибка инициализации, в том числе отсутствие инициализации; 2) не выводится результат, равный 0, или вместо 0 выводится неверное значение; 3) допущен выход за границу массива; 4) используется знак “<” вместо “<=”, “or” вместо “and” и т.п. 5) неверная работа с индексами элементов буфера.	
Не выполнены условия, позволяющие поставить 3 или 4 балла. Программа работает в целом верно, эффективно или нет, например, для решения задачи используется перебор всех возможных вариантов выбора элементов в парах. В реализации алгоритма допускается до трёх содержательных ошибок, допустимые виды ошибок перечислены в критериях на 3 балла. Количество синтаксических «описок» не должно быть более семи. Программа может быть неэффективна по времени, например, все числа запоминаются в массиве и перебираются все возможные произведения, т.е., по сути, реализовано решение задачи А без ограничений на количество введённых пар	2
Не выполнены условия, позволяющие поставить 2, 3 или 4 балла. Из описания алгоритма и общей структуры программы видно, что экзаменуемый в целом правильно представляет путь решения задачи. Допускается любое количество «описок». 1 балл ставится также за решения, верные лишь в частных случаях	1
Не выполнены критерии, позволяющие поставить 1, 2, 3 или 4 балла	0
<i>Максимальный балл для задания Б</i>	4
<i>Итоговый максимальный балл</i>	4