

**Сборник тренировочных материалов для подготовки
к государственной итоговой аттестации по ИНФОРМАТИКЕ и ИКТ
для слепых и поздноослепших обучающихся
по образовательным программам
СРЕДНЕГО общего образования**

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Тренировочные материалы предназначены для подготовки к единому государственному экзамену и государственному выпускному экзамену (в письменной форме) по информатике.

Тренировочные материалы состоят из пяти разделов, в которых задания сгруппированы следующим образом:

- раздел 1 – информация и её кодирование;
- раздел 2 – системы счисления, моделирование и компьютерный эксперимент, основы логики;
- раздел 3 – архитектура компьютеров и компьютерных сетей, обработка числовой информации, базы данных, поиск информации в сети;
- раздел 4 – алгоритмы, элементы теории алгоритмов;
- раздел 5 – программирование.

Раздел 1 содержит 10 заданий. Верное выполнение каждого из заданий 1–10 оценивается в 1 балл. Задания 1–10 могут включаться как в экзаменационные материалы ГВЭ-11, так и в КИМ ЕГЭ.

Раздел 2 содержит 14 заданий. За верное выполнение каждого из заданий 1–14 выставляется по 1 баллу. Задания 1–12 могут включаться как в экзаменационные материалы ГВЭ-11, так и в КИМ ЕГЭ, а задания 13 и 14 – только в КИМ ЕГЭ.

Раздел 3 содержит 9 заданий. Верное выполнение каждого из заданий 1–9 оценивается в 1 балл. Задания 1–9 могут включаться как в экзаменационные материалы ГВЭ-11, так и в КИМ ЕГЭ.

Раздел 4 содержит 19 заданий. За верное выполнение каждого из заданий 1–17 выставляется по 1 баллу. Ответ на задание 18 оценивается максимально в 2 балла, а ответ на задание 19 – в 3 балла в соответствии с критериями оценивания. Задания 1–17 могут включаться как в экзаменационные материалы ГВЭ-11, так и в КИМ ЕГЭ, задания 18, 19 используются только в КИМ ЕГЭ.

Раздел 5 содержит 8 заданий. За верное выполнение каждого из заданий 1–7 выставляется по 1 баллу. Ответ на задание 7 оценивается максимально в 3 балла, а на задание 8 – в 4 балла в соответствии с критериями оценивания. Задания 1–7 могут включаться как в экзаменационные материалы ГВЭ-11, так и в КИМ ЕГЭ, задания 7 и 8 используются только в КИМ ЕГЭ.

РАЗДЕЛ 1

Ответом к заданиям 1–10 является одна цифра или число. Ответ к заданиям 1–10 запишите в отведённом месте работы.

Информация и её кодирование

1

В некоторой информационной системе информация кодируется двоичными шестиразрядными словами. При передаче данных возможны их искажения, поэтому перед передачей в конец каждого слова добавляется седьмой (контрольный) разряд, таким образом, чтобы сумма разрядов нового слова, считая контрольный, была чётной. Например, к слову 110011 справа будет добавлен 0, а к слову 101100 – 1.

После приёма слова производится его обработка. При этом проверяется сумма его разрядов, включая контрольный. Если она нечётна, это означает, что при передаче этого слова произошёл сбой, и оно автоматически заменяется на зарезервированное слово 0000000. Если чётна, это означает, что сбоя не было или сбоев было больше одного. В этом случае слово не изменяется.

Исходное сообщение

1010101 0101011 0111010

было принято в виде

1010111 0101011 0111001.

Как будет выглядеть принятое сообщение после обработки?

- 1) 0000000 0101011 0111001
- 2) 1010111 0000000 0000000
- 3) 0000000 0101011 0000000
- 4) 1010111 0000000 0111001

Ответ:

2 Для передачи данных по каналу связи используется 5-битовый код. Сообщение содержит только буквы А, Б и В, которые кодируются следующими кодовыми словами:

А – 11011; Б – 01110; В – 10000.

При передаче возможны помехи. Однако некоторые ошибки можно попытаться исправить. Любые два из этих трёх кодовых слов отличаются друг от друга не менее чем в трёх позициях. Поэтому если при передаче слова произошла ошибка не более чем в одной позиции, то можно сделать обоснованное предположение о том, какая буква передавалась. (Говорят, что «код исправляет одну ошибку».) Например, если получено кодовое слово 11110, считается, что передавалась буква Б. (Отличие от кодового слова для Б только в одной позиции, для остальных кодовых слов отличий больше.) Если принятое кодовое слово отличается от кодовых слов для букв А, Б, В более чем в одной позиции, то считается, что произошла ошибка, которую нельзя исправить. Такая ошибка обозначается 'х'.

Получено сообщение 10010 11010 00110 01011. Декодируйте это сообщение – выберите правильный вариант.

- 1) хххх 2) ВАБА 3) ВхБх 4) ВАБх

Ответ:

3 По каналу связи передаются сообщения, каждое из которых содержит 16 букв А, 8 букв Б, 4 буквы В и 4 буквы Г (других букв в сообщениях нет). Каждую букву кодируют двоичной последовательностью. При выборе кода учитывались два требования:

а) ни одно кодовое слово не является началом другого (это нужно, чтобы код допускал однозначное декодирование);

б) общая длина закодированного сообщения должна быть как можно меньше.

Какой код из приведённых ниже следует выбрать для кодирования букв А, Б, В и Г?

- 1) А:0, Б:10, В:110, Г:111
2) А:0, Б:10, В:01, Г:11
3) А:1, Б:01, В:011, Г:001
4) А:00, Б:01, В:10, Г:11

Ответ:

4 Все 3-буквенные слова, составленные из букв Г, Е, П, А, Р, Д, записаны в алфавитном порядке и перенумерованы, начиная с 1.

Ниже приведено начало списка.

1. ААА
2. ААГ
3. ААД
4. ААЕ
5. ААП
6. ААР
7. АГА

Под каким номером в списке идёт первое слово, которое начинается с буквы Г?

Ответ: _____.

5 Все 3-буквенные слова, составленные из букв У, Ч, Е, Н, И, К, записаны в алфавитном порядке и перенумерованы, начиная с 1.

Ниже приведено начало списка.

1. ЕЕЕ
2. ЕЕИ
3. ЕЕК
4. ЕЕН
5. ЕЕУ
6. ЕЕЧ
7. ЕИЕ

Под каким номером в списке идёт первое слово, которое начинается с буквы К?

Ответ: _____.

6 Вася составляет 5-буквенные слова, в которых есть только буквы С, Л, О, Н, причём буква С используется в каждом слове ровно 1 раз. Каждая из других допустимых букв может встречаться в слове любое количество раз или не встречаться совсем. Словом считается любая допустимая последовательность букв, не обязательно осмысленная. Сколько существует таких слов, которые может написать Вася?

Ответ: _____.

7

Музыкальный фрагмент был оцифрован и записан в виде файла без использования сжатия данных. Получившийся файл был передан в город А по каналу связи за 30 секунд. Затем тот же музыкальный фрагмент был оцифрован повторно с разрешением в 2 раза выше и частотой дискретизации в 1,5 раза меньше, чем в первый раз. Сжатие данных не производилось. Полученный файл был передан в город Б; пропускная способность канала связи с городом Б в 4 раза выше, чем канала связи с городом А. Сколько секунд длилась передача файла в город Б? В ответе запишите только целое число, единицу измерения писать не нужно.

Ответ: _____.

8

Музыкальный фрагмент был записан в формате стерео (двухканальная запись), затем оцифрован и сохранён в виде файла без использования сжатия данных. Получившийся файл был передан в город А по каналу связи за 18 секунд. Затем тот же музыкальный фрагмент был повторно записан в формате моно и оцифрован с разрешением в 2 раза выше и частотой дискретизации в 2 раза больше, чем в первый раз. Сжатие данных не производилось. Полученный файл был передан в город Б; пропускная способность канала связи с городом Б в 5 раз ниже, чем канала связи с городом А. Сколько секунд длилась передача файла в город Б? В ответе запишите только целое число, единицу измерения писать не нужно.

Ответ: _____.

9

При регистрации в компьютерной системе каждому пользователю выдаётся пароль, состоящий из 15 символов и содержащий только символы из 8-символьного набора: А, В, С, D, E, F, G, H. В базе данных для хранения пароля отведено одинаковое и минимально возможное целое число байт. При этом используют посимвольное кодирование паролей, все символы кодируют одинаковым и минимально возможным количеством бит. Определите объём памяти (в байтах), необходимый для хранения паролей 20 пользователей. В ответе запишите только целое число – количество байт.

Ответ: _____.

10

При регистрации в компьютерной системе каждому пользователю выдаётся пароль, состоящий из 15 символов и содержащий только символы из 8-символьного набора: А, В, С, D, E, F, G, H. 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 24 байт на одного пользователя.

Определите объём памяти (в байтах), необходимый для хранения сведений о 30 пользователях. В ответе запишите только целое число – количество байт.

Ответ: _____.

Ответы к заданиям

№ задания	Ответ
1	1
2	2
3	1
4	37
5	73
6	405
7	10
8	180
9	120
10	900

РАЗДЕЛ 2

Ответом к заданиям 1–14 является одна цифра или число. Ответы к заданиям 1–14 запишите в отведённом месте работы.

Системы счисления

- 1** Сколько единиц в двоичной записи десятичного числа 513?
Ответ: _____.
- 2** Сколько единиц в двоичной записи десятичного числа 1023?
Ответ: _____.
- 3** Сколько единиц в двоичной записи восьмеричного числа 1731_8 ?
Ответ: _____.
- 4** Сколько единиц в двоичной записи шестнадцатеричного числа $11F0_{16}$?
Ответ: _____.
- 5** Сколько значащих нулей содержится в десятичной записи значения выражения:
 $100^{10} - 10^6 + 100$?
Ответ: _____.
- 6** Сколько единиц содержится в двоичной записи значения выражения:
 $4^{16} + 2^{36} - 16$?
Ответ: _____.
- 7** Значение арифметического выражения: $9^{12} + 3^7 - 27$ – записали в системе счисления с основанием 3. Сколько цифр «2» содержится в этой записи?
Ответ: _____.

Моделирование и компьютерный эксперимент

8 Между населёнными пунктами A, B, C, D, E, F построены дороги, протяжённость которых приведена в таблице. Отсутствие числа в таблице означает, что прямой дороги между пунктами нет.

	A	B	C	D	E	F
A		2				12
B	2		2	4		
C		2		3		
D		4	3		3	5
E				3		1
F	12			5	1	

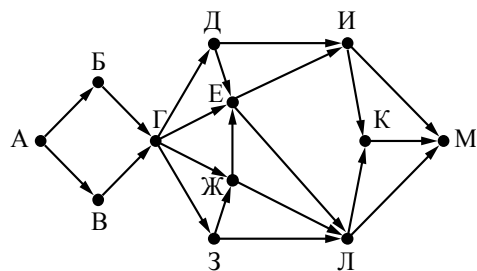
Определите длину кратчайшего пути между пунктами A и F при условии, что передвигаться можно только по указанным в таблице дорогам.

Ответ: _____.

9 На рисунке представлена схема дорог, связывающих города A, Б, В, Г, Д, Е, Ж, З, И, К, Л, М.

По каждой дороге можно двигаться только в одном направлении, указанном стрелкой.

Сколько существует различных путей из города A в город M?



Ответ: _____.

Основы логики

10 Слова заполнял таблицу истинности для выражения F. Он успел заполнить лишь небольшой фрагмент таблицы.

x1	x2	x3	x4	x5	x6	x7	x8	F
	1				0	0		0
1		1					0	1
		0						1

Каким выражением может быть F?

- 1) $\neg x1 \vee \neg x2 \vee x3 \vee \neg x4 \vee \neg x5 \vee x6 \vee x7 \vee \neg x8$
- 2) $\neg x1 \wedge x2 \wedge \neg x3 \wedge x4 \wedge x5 \wedge x6 \wedge x7 \wedge x8$
- 3) $\neg x1 \vee \neg x2 \vee x3 \vee \neg x4 \vee \neg x5 \vee \neg x6 \vee x7 \vee \neg x8$
- 4) $x1 \wedge \neg x2 \wedge x3 \wedge \neg x4 \wedge x5 \wedge x6 \wedge \neg x7 \wedge \neg x8$

Ответ:

11 На числовой прямой даны два отрезка: $D = [15; 40]$ и $C = [21; 63]$. Укажите наименьшую возможную длину такого отрезка A, что формула

$$(x \in D) \rightarrow ((\neg(x \in C) \wedge \neg(x \in A)) \rightarrow \neg(x \in D))$$

истинна (т.е. принимает значение 1 при любом значении переменной x).

Ответ: _____.

12 Обозначим через $\text{ДЕЛ}(n, m)$ утверждение «натуральное число n делится без остатка на натуральное число m».

Для какого наибольшего натурального числа A формула

$$\neg \text{ДЕЛ}(x, A) \rightarrow (\text{ДЕЛ}(x, 6) \rightarrow \neg \text{ДЕЛ}(x, 4))$$

тождественно истинна (т.е. принимает значение 1 при любом натуральном значении переменной x)?

Ответ: _____.

13

Обозначим через $m \& n$ поразрядную конъюнкцию неотрицательных целых чисел m и n . Так, например, $14 \& 5 = 1110_2 \& 0101_2 = 0100_2 = 4$.

Для какого наибольшего неотрицательного целого числа A формула

$$x \& A \neq 0 \rightarrow (x \& 12 = 0 \rightarrow x \& 5 \neq 0)$$

тождественно истинна (т.е. принимает значение 1 при любом неотрицательном целом значении переменной x)?

Ответ: _____.

14

Сколько существует различных наборов значений логических переменных $x_1, x_2, \dots, x_6, y_1, y_2, \dots, y_6$, которые удовлетворяют всем перечисленным ниже условиям?

$$(x_1 \vee y_1) \rightarrow (x_2 \wedge y_2) = 1$$

$$(x_2 \vee y_2) \rightarrow (x_3 \wedge y_3) = 1$$

...

$$(x_5 \vee y_5) \rightarrow (x_6 \wedge y_6) = 1$$

В ответе **не нужно** перечислять все различные наборы значений переменных $x_1, x_2, \dots, x_6, y_1, y_2, \dots, y_6$, при которых выполнена данная система равенств. В качестве ответа Вам нужно указать количество таких наборов.

Ответ: _____.

Ответы к заданиям

№ задания	Ответ
1	2
2	10
3	7
4	6
5	5
6	29
7	4
8	10
9	48
10	1
11	6
12	12
13	13
14	19

РАЗДЕЛ 3

Ответом к заданиям 1–9 является одна цифра или число. Ответы к заданиям 1–9 запишите в отведённом месте работы.

Архитектура компьютеров и компьютерных сетей

1 В терминологии сетей TCP/IP маской сети называется двоичное число, определяющее, какая часть IP-адреса узла сети относится к адресу сети, а какая – к адресу самого узла в этой сети. Обычно маска записывается по тем же правилам, что и IP-адрес, – в виде четырёх байтов, причём каждый байт записывается в виде десятичного числа. При этом в маске сначала (в старших разрядах) стоят единицы, а затем с некоторого разряда – нули. Адрес сети получается в результате применения поразрядной конъюнкции к заданному IP-адресу узла и маске.
Например, если IP-адрес узла равен 231.32.255.131, а маска равна 255.255.240.0, то адрес сети равен 231.32.240.0.

Для узла с IP-адресом 111.81.176.127 адрес сети равен 111.81.160.0. Чему равен третий слева байт маски? Ответ запишите в виде десятичного числа.

Ответ: _____.

2 В терминологии сетей TCP/IP маской сети называется двоичное число, определяющее, какая часть IP-адреса узла сети относится к адресу сети, а какая – к адресу самого узла в этой сети. Обычно маска записывается по тем же правилам, что и IP-адрес, – в виде четырёх байтов, причём каждый байт записывается в виде десятичного числа. При этом в маске сначала (в старших разрядах) стоят единицы, а затем с некоторого разряда – нули. Адрес сети получается в результате применения поразрядной конъюнкции к заданному IP-адресу узла и маске.
Например, если IP-адрес узла равен 231.32.255.131, а маска равна 255.255.240.0, то адрес сети равен 231.32.240.0.

Для узла с IP-адресом 117.191.224.137 адрес сети равен 117.191.192.0. Чему равно наименьшее возможное значение третьего слева байта маски? Ответ запишите в виде десятичного числа.

Ответ: _____.

Обработка числовой информации

3 Дан фрагмент электронной таблицы. Из ячейки A2 в ячейку B1 была скопирована формула. При копировании адреса ячеек в формуле автоматически изменились. Каким стало числовое значение формулы в ячейке B1?

	A	B	C	D	E
1	40		400	4000	40000
2	=C\$2 + \$D3	3	300	3000	30000
3	20	2	200	2000	20000
4	10	1	100	1000	10000

Примечание: знак \$ обозначает абсолютную адресацию.

Ответ: _____.

4 Дан фрагмент электронной таблицы. Из ячейки E4 в ячейку D3 была скопирована формула. При копировании адреса ячеек в формуле автоматически изменились. Каким стало числовое значение формулы в ячейке D3?

	A	B	C	D	E
1	40	5	400	70	4
2	30	6	300	60	3
3	20	7	200		2
4	10	8	100	40	=B3 * \$C\$2

Примечание: знак \$ обозначает абсолютную адресацию.

Ответ: _____.

Базы данных

5

Ниже представлены две таблицы из базы данных. Каждая строка таблицы 2 содержит информацию о ребёнке и об одном из его родителей. Информация представлена значением поля ID в соответствующей строке таблицы 1. Определите на основании приведённых данных, сколько потомков мужского пола у Гуревича И.Т.

Таблица 1		
ID	Фамилия_И.О.	Пол
85	Гуревич И.Т.	М
82	Гуревич А.И.	М
42	Цейс А.Т.	Ж
71	Петров Т.М.	М
23	Петров А.Т.	М
13	Цейс И.И.	Ж
95	Черных Т.Н.	Ж
10	Черных Н.И.	М
	...	

Таблица 2	
ID_Родителя	ID_Ребёнка
95	82
85	13
71	42
85	82
13	42
71	23
13	23
95	13
85	10
...	...

Ответ:

6

Ниже представлены две таблицы из базы данных. Каждая строка таблицы 2 содержит информацию о ребёнке и об одном из его родителей. Информация представлена значением поля ID в соответствующей строке таблицы 1. Определите на основании приведённых данных фамилию и инициалы племянницы Лещенко П.И.

Пояснение: племянницей считается дочь брата или сестры.

Таблица 1		
ID	Фамилия_И.О.	Пол
14	Лещенко Н.А.	Ж
23	Лещенко И.П.	М
25	Лещенко П.И.	М
26	Лещенко П.П.	М
34	Робертс А.И.	Ж
35	Робертс В.С.	Ж
36	Робертс С.С.	М
44	Дрозд А.С.	Ж
45	Дрозд В.А.	М
46	Иванец О.С.	М
47	Иванец П.О.	М
54	Челидзе А.П.	Ж
64	Саркисян П.А.	Ж
...

Таблица 2	
ID_Родителя	ID_Ребёнка
23	25
44	25
25	26
64	26
23	34
44	34
34	35
36	35
14	36
34	46
36	46
25	54
64	54
...	...

- 1) Робертс В.С.
- 2) Иванец О.С.
- 3) Челидзе А.П.
- 4) Лещенко Н.А.

Ответ:

7

Поиск информации в сети

В языке запросов поискового сервера для обозначения логической операции «ИЛИ» используется символ «|», а для обозначения логической операции «И» – символ «&».

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

Запрос	Найдено страниц (в сотнях тысяч)
<i>Поле</i>	45
<i>Пшеница</i>	24
<i>Напряжённость</i>	44
<i>Напряжённость Поле Пшеница</i>	74
<i>Напряжённость & Поле</i>	19
<i>Напряжённость & Пшеница</i>	0

Какое количество страниц (в сотнях тысяч) будет найдено по запросу *Поле & Пшеница*?

Считается, что все запросы выполнялись практически одновременно, так что набор страниц, содержащих все искомые слова, не изменялся за время выполнения запросов.

Ответ: _____.

8

В языке запросов поискового сервера для обозначения логической операции «ИЛИ» используется символ «|», а для обозначения логической операции «И» – символ «&».

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

Запрос	Найдено страниц (в сотнях тысяч)
<i>Поле</i>	45
<i>Пшеница</i>	24
<i>Напряжённость</i>	44
<i>Поле & Пшеница</i>	17
<i>Напряжённость & Поле</i>	19
<i>Напряжённость & Пшеница</i>	0

Какое количество страниц (в сотнях тысяч) будет найдено по запросу *Напряжённость | Поле | Пшеница*?

Считается, что все запросы выполнялись практически одновременно, так что набор страниц, содержащих все искомые слова, не изменялся за время выполнения запросов.

Ответ: _____.

9

В языке запросов поискового сервера для обозначения логической операции «ИЛИ» используется символ «|», а для обозначения логической операции «И» – символ «&».

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

Запрос	Найдено страниц (в тысячах)
<i>Пушкин & Лермонтов</i>	320
<i>Пушкин & Гоголь</i>	280
<i>Пушкин & (Лермонтов Гоголь)</i>	520

Какое количество страниц (в тысячах) будет найдено по запросу *Пушкин & Лермонтов & Гоголь*?

Считается, что все запросы выполнялись практически одновременно, так что набор страниц, содержащих все искомые слова, не изменялся за время выполнения запросов.

Ответ: _____.

Ответы к заданиям

№ задания	Ответ
1	224
2	192
3	6000
4	9000
5	3
6	1
7	20
8	77
9	80

РАЗДЕЛ 4

Ответом к заданиям 1–17 является одна цифра или число. Ответы к заданиям 1–17 запишите в отведённом месте работы.

При выполнении заданий 18 и 19 используйте дополнительный лист. Ответ пишите чётко и разборчиво.

Алгоритмы. Элементы теории алгоритмов

1 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, 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 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 минимальное число R , которое превышает 43 и может являться результатом работы алгоритма. В ответе это число запишите в десятичной системе.

Ответ: _____.

2

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, 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 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 алгоритма больше 45. В ответе это число запишите в десятичной системе счисления.

Ответ: _____.

3

У исполнителя Калькулятор две команды, которым присвоены номера:

1. прибавь 2,

2. умножь на 5.

Выполняя первую из них, Калькулятор прибавляет к числу на экране 2, а выполняя вторую, умножает его на 5.

Так, для программы

умножь на 5,

прибавь 2,

умножь на 5,

прибавь 2,

нужно написать: 2121. Эта программа преобразует число 1 в число 37.

Запишите программу, которая преобразует **число 2 в число 24** и содержит не более четырёх команд. Указывайте лишь номера команд. Если таких программ более одной, то запишите любую из них.

Ответ: _____.

4

У исполнителя Аккорд две команды, которым присвоены номера:

1. **вычти 1,**
2. **умножь на 5.**

Выполняя первую из них, Аккорд вычитает от числа на экране 1, а выполняя вторую, умножает это число на 5.

Так, для программы

умножь на 5,

вычти 1,

вычти 1

нужно написать: 211. Эта программа преобразует число 5 в число 23.

Запишите программу, которая преобразует **число 1 в число 99** и содержит не более пяти команд. Указывайте лишь номера команд. Если таких программ более одной, то запишите любую из них.

Ответ: _____.

5

Исполнитель Чертёжник перемещается на координатной плоскости, оставляя след в виде линии. Чертёжник может выполнять команду **сместиться на (a, b)** , где a, b – целые числа. Эта команда перемещает Чертёжника из точки с координатами (x, y) в точку с координатами $(x + a; y + b)$.

Например, если Чертёжник находится в точке с координатами $(4, 2)$, то команда **сместиться на $(2, -3)$** переместит Чертёжника в точку $(6, -1)$.

Цикл

ПОВТОРИ число РАЗ

последовательность команд

КОНЕЦ ПОВТОРИ

означает, что *последовательность команд* будет выполнена указанное *число* раз (число должно быть натуральным).

Чертёжнику был дан для исполнения следующий алгоритм (количество повторений и величины смещения в первой из повторяемых команд неизвестны):

НАЧАЛО

сместиться на $(-12, 11)$

ПОВТОРИ ... РАЗ

сместиться на $(..., ...)$

сместиться на $(-6, 3)$

КОНЕЦ ПОВТОРИ

сместиться на $(-13, -26)$

КОНЕЦ

В результате выполнения этого алгоритма Чертёжник возвращается в исходную точку. Какое наибольшее число повторений могло быть указано в конструкции «ПОВТОРИ ... РАЗ»?

Ответ: _____.

6

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

А) заменить (v, w).

Эта команда заменяет в строке первое слева вхождение цепочки v на цепочку w . Например, выполнение команды

заменить (111, 27)

преобразует строку 05111150 в строку 0527150.

Если в строке нет вхождений цепочки v , то выполнение команды **заменить** (v, w) не меняет эту строку.

Б) 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 исполнителя при этом не изменяется.

Цикл

ПОКА *условие*

последовательность команд

КОНЕЦ ПОКА

выполняется, пока условие истинно.

В конструкции

ЕСЛИ *условие*

ТО команда1

ИНАЧЕ команда2

КОНЕЦ ЕСЛИ

выполняется *команда1* (если условие истинно) или *команда2* (если условие ложно).

Какая строка получится в результате применения приведённой ниже программы к строке, состоящей из 84 идущих подряд цифр 8? В ответе запишите полученную строку.

НАЧАЛО

ПОКА **нашлось** (777) ИЛИ **нашлось** (888)

ЕСЛИ **нашлось** (777)

ТО заменить (777, 8)

ИНАЧЕ заменить (888, 7)

КОНЕЦ ЕСЛИ

КОНЕЦ ПОКА

КОНЕЦ

Ответ: _____.

7

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

А) заменить (v, w).

Эта команда заменяет в строке первое слева вхождение цепочки v на цепочку w . Например, выполнение команды

заменить (111, 27)

преобразует строку 05111150 в строку 0527150.

Если в строке нет вхождений цепочки v , то выполнение команды **заменить** (v, w) не меняет эту строку.

Б) 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 исполнителя при этом не изменяется.

Цикл

ПОКА *условие*

последовательность команд

КОНЕЦ ПОКА

выполняется, пока условие истинно.

В конструкции

ЕСЛИ *условие*

ТО команда1

ИНАЧЕ команда2

КОНЕЦ ЕСЛИ

выполняется *команда1* (если условие истинно) или *команда2* (если условие ложно).

Какая строка получится в результате применения приведённой ниже программы к строке, состоящей из 95 идущих подряд цифр 3? В ответе запишите полученную строку.

НАЧАЛО

ПОКА **нашлось** (999) ИЛИ **нашлось** (333)

ЕСЛИ **нашлось** (999)

ТО заменить (999, 3)

ИНАЧЕ заменить (333, 9)

КОНЕЦ ЕСЛИ

КОНЕЦ ПОКА

КОНЕЦ

Ответ: _____.

8

Ниже на пяти языках программирования записаны две рекурсивные функции (процедуры): F и G.

Бейсик	Python
<pre> DECLARE SUB F(n) DECLARE SUB G(n) SUB F(n) PRINT "*" IF n > 0 THEN G(n - 1) END SUB SUB G(n) PRINT "*" IF n > 0 THEN F(n - 1) END SUB </pre>	<pre> def F(n): print("*") if n > 0: G(n - 1) def G(n): print("*") if n > 0: F(n - 1) </pre>
Алгоритмический язык	Паскаль
<pre> алг F(цел n) нач вывод "*" если n > 0 то G(n - 1) все кон алг G(цел n) нач вывод "*" если n > 0 то F(n - 1) все кон </pre>	<pre> procedure F(n: integer); forward; procedure G(n: integer); forward; procedure F(n: integer); begin writeln('*'); if n > 0 then G(n - 1); end; procedure G(n: integer); begin writeln('*'); if n > 0 then F(n - 1); end; </pre>
Си	
<pre> void F(int n); void G(int n); void F(int n){ printf("*"); if (n > 0) G(n - 1); } void G(int n){ printf("*"); if (n > 0) F(n - 1); } </pre>	

Сколько символов «звёздочка» будет напечатано на экране при выполнении вызова F(14)?

Ответ: _____.

27

9

Ниже на пяти языках программирования записана рекурсивная функция (процедура) F.

Бейсик	Python
<pre> SUB F(n) print n, IF n >= 7 THEN F(n - 3) F(n - 1) END IF END SUB </pre>	<pre> def F(n): print(n, end='') if n >= 7: F(n - 3) F(n - 1) </pre>
Алгоритмический язык	Паскаль
<pre> алг F(цел n) нач вывод n если n >= 7 то F(n - 3) F(n - 1) все кон </pre>	<pre> procedure F(n: integer); begin write(n); if n >= 7 then begin F(n - 3); F(n - 1) end end; </pre>
Си	
<pre> void F(int n) { printf("%d", n); if (n >= 7) { F(n - 3); F(n - 1); } } </pre>	

Что выведет программа при вызове F(9)? В ответе запишите последовательность выведенных цифр слитно (без пробелов).

Ответ: _____.

28

10

В программе используется одномерный целочисленный массив A с индексами от 0 до 9. Значения элементов равны 9, 8, 7, 6, 5, 4, 3, 2, 1, 0 соответственно, т.е. $A[0] = 9$, $A[1] = 8$ и т.д.

Определите значение переменной s после выполнения следующего фрагмента этой программы (записанного ниже на разных языках программирования).

Бейсик	Python
<pre>s = 0 FOR j = 0 TO 8 IF A(j) > A(j+1) THEN s = s + 1 t = A(j) A(j) = A(j+1) A(j+1) = t END IF NEXT j</pre>	<pre>s = 0 for j in range(9): if A[j] > A[j+1]: s = s + 1 t = A[j] A[j] = A[j+1] A[j+1] = t</pre>
Алгоритмический язык	Паскаль
<pre>s := 0 нц для j от 0 до 8 если A[j] > A[j+1] то s := s + 1 t := A[j] A[j] := A[j+1] A[j+1] := t все кц</pre>	<pre>s := 0; for j := 0 to 8 do if A[j] > A[j+1] then begin s := s + 1; t := A[j]; A[j] := A[j+1]; A[j+1] := t; end;</pre>
Си	
<pre>s = 0; for (j = 0; j < 9; j++) if (A[j] > A[j+1]) { s++; t = A[j]; A[j] = A[j+1]; A[j+1] = t; }</pre>	

Ответ: _____.

11

В программе используется одномерный целочисленный массив A с индексами от 0 до 9. Значения элементов равны 4, 7, 3, 8, 5, 0, 1, 2, 9, 6 соответственно, т.е. $A[0] = 4$, $A[1] = 7$ и т.д.

Определите значение переменной c после выполнения следующего фрагмента этой программы (записанного ниже на разных языках программирования).

Бейсик	Python
<pre>c = 0 FOR i = 1 TO 9 IF A(i) < A(0) THEN c = c + 1 t = A(i) A(i) = A(0) A(0) = t ENDIF NEXT i</pre>	<pre>c = 0 for i in range(1,10): if A[i] < A[0]: c = c + 1 t = A[i] A[i] = A[0] A[0] = t</pre>
Алгоритмический язык	Паскаль
<pre>c := 0 нц для i от 1 до 9 если A[i] < A[0] то c := c + 1 t := A[i] A[i] := A[0] A[0] := t все кц</pre>	<pre>c := 0; for i := 1 to 9 do if A[i] < A[0] then begin c := c + 1; t := A[i]; A[i] := A[0]; A[0] := t; end;</pre>
Си	
<pre>c = 0; for (i = 1; i < 10; i++) if (A[i] < A[0]) { c++; t = A[i]; A[i] = A[0]; A[0] = t; }</pre>	

Ответ: _____.

12

Ниже на пяти языках программирования записан алгоритм. Получив на вход число x , этот алгоритм печатает два числа: L и M . Укажите **наибольшее** число x , при вводе которого алгоритм печатает сначала 5, а потом 7.

Бейсик	Python
<pre>DIM X, L, M, Q AS INTEGER INPUT X Q = 8 L = 0 WHILE X >= Q L = L + 1 X = X - Q WEND M = X IF M < L THEN M = L L = X END IF PRINT L PRINT M</pre>	<pre>x = int(input()) Q = 8 L = 0 while x >= Q: L = L + 1 x = x - Q M = x if M < L: M = L L = x print(L) print(M)</pre>
Алгоритмический язык	Паскаль
<pre>алг нач цел x, L, M, Q ввод x Q := 8 L := 0 нц пока x >= Q L := L + 1 x := x - Q кц M := x если M < L то M := L L := x все вывод L, M кон</pre>	<pre>var x, L, M, Q: integer; begin readln(x); Q := 8; L := 0; while x >= Q do begin L := L + 1; x := x - Q; end; M := x; if M < L then begin M := L; L := x; end; writeln(L); writeln(M); end.</pre>

Си

```
#include<stdio.h>
void main()
{
    int x, L, M, Q;
    scanf("%d", &x);
    Q = 8;
    L = 0;
    while (x >= Q){
        L = L + 1;
        x = x - Q;
    }
    M = x;
    if (M < L){
        M = L;
        L = x;
    }
    printf("%d\n%d", L, M);
}
```

Ответ: _____.

13

Ниже на пяти языках программирования записан алгоритм. Получив на вход число x , этот алгоритм печатает два числа: L и M . Укажите **наибольшее** число x , при вводе которого алгоритм печатает сначала 7, а потом 8.

Бейсик	Python
<pre>DIM X, L, M AS INTEGER INPUT X L = 0 M = 0 WHILE X > 0 M = M + 1 IF X MOD 2 <> 0 THEN L = L + 1 ENDIF X = X \ 2 WEND PRINT L PRINT M</pre>	<pre>x = int(input()) L = 0 M = 0 while x > 0: M = M + 1 if x % 2 != 0: L = L + 1 x = x // 2 print(L) print(M)</pre>

Алгоритмический язык	Паскаль
<pre> алг нач цел x, L, M ввод x L := 0 M := 0 нц пока x > 0 M := M + 1 если mod(x,2) <> 0 то L := L + 1 все x := div(x,2) кц вывод L, M кон </pre>	<pre> var x, L, M: integer; begin readln(x); L := 0; M := 0; while x>0 do begin M := M+1; if x mod 2 <> 0 then L := L + 1; x := x div 2; end; writeln(L); writeln(M); end. </pre>
Си <pre> #include<stdio.h> void main() { int x, L, M; scanf("%d", &x); L = 0; M = 0; while (x > 0){ M = M + 1; if(x % 2 != 0){ L = L + 1; } x = x / 2; } printf("%d\n%d", L, M); } </pre>	

Ответ: _____.

14

Ниже на пяти языках программирования записан алгоритм. Получив на вход число x , этот алгоритм печатает число M . Известно, что $x > 50$. Укажите **наименьшее** такое (т.е. большее 50) число x , при вводе которого алгоритм печатает 14.

Бейсик	Python
<pre> DIM X, L, M AS INTEGER INPUT X L = X M = 35 IF L MOD 2 = 0 THEN M = 28 ENDIF WHILE L <> M IF L > M THEN L = L - M ELSE M = M - L ENDIF WEND PRINT M </pre>	<pre> x = int(input()) L = x M = 35 if L % 2 == 0: M = 28 while L != M: if L > M: L = L - M else: M = M - L print(M) </pre>
Алгоритмический язык	Паскаль
<pre> алг нач цел x, L, M ввод x L := x M := 35 если mod(L,2)=0 то M := 28 все нц пока L <> M если L > M то L := L - M иначе M := M - L кц вывод M кон </pre>	<pre> var x, L, M: integer; begin readln(x); L := x; M := 35; if L mod 2 = 0 then M := 28; while L <> M do if L > M then L := L - M; else M := M - L; end; end; writeln(M); end. </pre>

Си

```
#include<stdio.h>
void main()
{
    int x, L, M;
    scanf("%d", &x);
    L = x;
    M = 35;
    if (L % 2 == 0)
        M = 28;
    while (L != M){
        if(L > M)
            L = L - M;
        else
            M = M - L;
    }
    printf("%d", M);
}
```

Ответ: _____.

15

Исполнитель Плюс преобразует число на экране.

У исполнителя есть две команды, которым присвоены номера:

1. Прибавить 2

2. Прибавить 4

Первая команда увеличивает число на экране на 2, вторая увеличивает это число на 4. Программа для исполнителя Плюс – это последовательность команд.

Сколько существует программ, которые число 9 преобразуют в число 25?

Ответ: _____.

16

Исполнитель Апрель15 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 Апрель15 – это последовательность команд.

Сколько существует программ, для которых при исходном числе 1 результатом является число 21 и при этом траектория вычислений содержит число 10?

Траектория вычислений программы – это последовательность результатов выполнения всех команд программы. Например, для программы **121** при исходном числе 7 траектория будет состоять из чисел 8, 16, 17.

Ответ: _____.

17

Исполнитель Май15 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 Май15 – это последовательность команд.

Сколько существует программ, для которых при исходном числе 1 результатом является число 21 и при этом траектория вычислений содержит число 10 и не содержит числа 15?

Траектория вычислений программы – это последовательность результатов выполнения всех команд программы. Например, для программы **121** при исходном числе 7 траектория будет состоять из чисел 8, 16, 17.

Ответ: _____.

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	Python
<pre>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</pre>	<pre># допускается также # использовать две # целочисленные переменные j и k a = [] n = 20 for i in range(0, n): a.append(int(input())) ...</pre>
Алгоритмический язык	Паскаль
<pre><u>алг</u> <u>нач</u> цел N = 20 целтаб a[1:N] цел i, j, k нц для i от 1 до N ввод a[i] кц ... <u>кон</u></pre>	<pre>const N = 20; var a: array [1..N] of integer; i, j, k: integer; begin for i := 1 to N do readln(a[i]); ... end.</pre>

Си	Естественный язык
<pre>#include <stdio.h> #define N 20 int main() { int a[N]; int i, j, k; for (i = 0; i < N; i++) scanf("%d", &a[i]); ... return 0; }</pre>	<p>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. ...</p>

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).

Два игрока, Петя и Ваня, играют в следующую игру. Перед игроками лежат две кучи камней. Игроки ходят по очереди, первый ход делает Петя. За один ход игрок может добавить в одну из куч (по своему выбору) **один** камень или увеличить количество камней в куче в **два раза**. Например, пусть в одной куче 10 камней, а в другой 7 камней; такую позицию в игре будем обозначать (10, 7). Тогда за один ход можно получить любую из четырёх позиций: (11, 7), (20, 7), (10, 8), (10, 14). Для того чтобы делать ходы, у каждого игрока есть неограниченное количество камней.

Игра завершается в тот момент, когда суммарное количество камней в кучах становится не менее 55. Победителем считается игрок, сделавший последний ход, т.е. первым получивший такую позицию, что в кучах всего будет 55 или больше камней.

Будем говорить, что игрок имеет *выигрышную стратегию*, если он может выиграть при любых ходах противника. Описать стратегию игрока – значит описать, какой ход он должен сделать в любой ситуации, которая ему может встретиться при различной игре противника. Например, при начальных позициях (4, 26) и (5, 25) выигрышная стратегия есть у Пети. Чтобы выиграть, ему достаточно удвоить количество камней во второй куче.

Задание 1. Для каждой из начальных позиций (4, 25), (6, 24) укажите, кто из игроков имеет выигрышную стратегию. В каждом случае опишите выигрышную стратегию; объясните, почему эта стратегия ведёт к выигрышу, и укажите, какое наибольшее количество ходов может потребоваться победителю для выигрыша при этой стратегии.

Задание 2. Для каждой из начальных позиций (4, 24), (5, 24), (6, 23) укажите, кто из игроков имеет выигрышную стратегию. В каждом случае опишите выигрышную стратегию; объясните, почему эта стратегия ведёт к выигрышу, и укажите, какое наибольшее количество ходов может потребоваться победителю для выигрыша при этой стратегии.

Задание 3. Для начальной позиции (5, 23) укажите, кто из игроков имеет выигрышную стратегию. Опишите выигрышную стратегию; объясните, почему эта стратегия ведёт к выигрышу, и укажите, какое наибольшее количество ходов может потребоваться победителю для выигрыша при этой стратегии. Постройте дерево всех партий, возможных при указанной Вами выигрышной стратегии. Представьте дерево в виде рисунка или таблицы.

Ответы к заданиям

№ задания	Ответ
1	46
2	11
3	1211
4	21221
5	5
6	78
7	993
8	15
9	9685746
10	9
11	2
12	61
13	254
14	70
15	34
16	28
17	14

Критерии оценивания заданий с развёрнутым ответом

18

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	Python
<pre>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 ...</pre>	<pre># допускается также # использовать две # целочисленные переменные j и k a = [] n = 20 for i in range(0, n): a.append(int(input())) ...</pre>
Алгоритмический язык	Паскаль
<pre><u>алг</u> <u>нач</u> цел N = 20 целтаб a[1:N] цел i, j, k нц для i от 1 до N ввод a[i] кц ... <u>кон</u></pre>	<pre>const N = 20; var a: array [1..N] of integer; i, j, k: integer; begin for i := 1 to N do readln(a[i]); ... end.</pre>

Си	Естественный язык
<pre>#include <stdio.h> #define N 20 int main() { int a[N]; int i, j, k; for (i = 0; i<N; i++) scanf("%d", &a[i]); ... return 0; }</pre>	<p>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. ...</p>

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).

Содержание верного ответа и указания по оцениванию (допускаются иные формулировки решений, приводящие к правильному результату)
На языке Паскаль
<pre>k := 0; for i := 1 to N-1 do if (a[i] mod 3=0) or (a[i+1] mod 3=0) then inc(k); writeln(k);</pre>
На алгоритмическом языке
<pre>k := 0; нц для i от 1 до N-1 если mod(a[i], 3)=0 или mod(a[i+1], 3)=0 то k := k+1 все кц вывод k</pre>
На языке Бейсик
<pre>K = 0 FOR I = 1 TO N-1 IF (A(I) MOD 3 = 0) OR (A(I + 1) MOD 3 = 0) THEN K = K+1 END IF NEXT I PRINT K</pre>

На языке Си	
<pre>k = 0; for (i = 0; i < N - 1; i++) if (a[i] % 3 == 0 a[i + 1] % 3 == 0) k++; printf("%d", k);</pre>	
На языке Python	
<pre>k = 0 for i in range(0, n - 1): if (a[i] % 3 == 0 or a[i + 1] % 3 == 0): k += 1 print(k)</pre>	
На естественном языке	
<p>Записываем в переменную <i>K</i> начальное значение, равное 0. В цикле от первого элемента до предпоследнего находим остаток от деления текущего и следующего элемента массива на 3. Если первый или второй из полученных остатков равен 0, увеличиваем переменную <i>K</i> на единицу. После завершения цикла выводим значение переменной <i>K</i></p>	
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 (например, используется цикл от 1 до <i>N</i>); 2) не инициализируется или неверно инициализируется счётчик количества найденных пар; 3) счётчик количества пар в цикле не изменяется или изменяется неверно; 4) неверно проверяется делимость на 3; 5) на делимость проверяются не сами элементы, а их индексы; 6) при проверке выполнения условия для пары элементов используются неверные индексы; 7) в сложном логическом условии простые проверки верны, но условие в целом построено неверно (например, перепутаны операции «И» и «ИЛИ», неверно расставлены скобки в логическом выражении); 8) отсутствует вывод ответа; 9) используется переменная, не объявленная в разделе описания переменных; 10) не указано или неверно указано условие завершения цикла; 11) индексная переменная в цикле не меняется (например, в цикле <code>while</code>) или меняется неверно; 12) неверно расставлены операторные скобки 	1
<p>Ошибок, перечисленных в п. 1–12, две или больше, или алгоритм сформулирован неверно (в том числе при отсутствии цикла в явном или неявном виде)</p>	0
<i>Максимальный балл</i>	2

Два игрока, Петя и Ваня, играют в следующую игру. Перед игроками лежат две кучи камней. Игроки ходят по очереди, первый ход делает Петя. За один ход игрок может добавить в одну из куч (по своему выбору) **один** камень или увеличить количество камней в куче в **два раза**. Например, пусть в одной куче 10 камней, а в другой 7 камней; такую позицию в игре будем обозначать (10, 7). Тогда за один ход можно получить любую из четырёх позиций: (11, 7), (20, 7), (10, 8), (10, 14). Для того чтобы делать ходы, у каждого игрока есть неограниченное количество камней.

Игра завершается в тот момент, когда суммарное количество камней в кучах становится не менее 55. Победителем считается игрок, сделавший последний ход, т.е. первым получивший такую позицию, что в кучах всего будет 55 или больше камней.

Будем говорить, что игрок имеет *выигрышную стратегию*, если он может выиграть при любых ходах противника. Описать стратегию игрока – значит описать, какой ход он должен сделать в любой ситуации, которая ему может встретиться при различной игре противника. Например, при начальных позициях (4, 26) и (5, 25) выигрышная стратегия есть у Пети. Чтобы выиграть, ему достаточно удвоить количество камней во второй куче.

Задание 1. Для каждой из начальных позиций (4, 25), (6, 24) укажите, кто из игроков имеет выигрышную стратегию. В каждом случае опишите выигрышную стратегию; объясните, почему эта стратегия ведёт к выигрышу, и укажите, какое наибольшее количество ходов может потребоваться победителю для выигрыша при этой стратегии.

Задание 2. Для каждой из начальных позиций (4, 24), (5, 24), (6, 23) укажите, кто из игроков имеет выигрышную стратегию. В каждом случае опишите выигрышную стратегию; объясните, почему эта стратегия ведёт к выигрышу, и укажите, какое наибольшее количество ходов может потребоваться победителю для выигрыша при этой стратегии.

Задание 3. Для начальной позиции (5, 23) укажите, кто из игроков имеет выигрышную стратегию. Опишите выигрышную стратегию; объясните, почему эта стратегия ведёт к выигрышу, и укажите, какое наибольшее количество ходов может потребоваться победителю для выигрыша при этой стратегии. Постройте дерево всех партий, возможных при указанной Вами выигрышной стратегии. Представьте дерево в виде рисунка или таблицы.

Содержание верного ответа и указания по оцениванию

(допускаются иные формулировки ответа, не искажающие его смысла)

Задание 1. В начальных позициях (4, 25), (6, 24) выигрышная стратегия есть у Вани. При начальной позиции (4, 25) после первого хода Пети может получиться одна из следующих четырёх позиций: (5, 25), (8, 25), (4, 26), (4, 50). Каждая из этих позиций содержит менее 55 камней. При этом из любой из этих позиций Ваня может получить позицию, содержащую не менее 55 камней, удвоив количество камней во второй куче. Для позиции (6, 24) после первого хода Пети может получиться одна из следующих четырёх позиций: (7, 24), (12, 24), (6, 25), (6, 48). Каждая из этих позиций содержит менее 55 камней. При этом из любой из этих позиций Ваня может получить позицию, содержащую не менее 55 камней, удвоив количество камней во второй куче. Таким образом, Ваня при любом ходе Пети выигрывает своим первым ходом.

Задание 2. В начальных позициях (4, 24), (5, 24) и (6, 23) выигрышная стратегия есть у Пети. При начальной позиции (4, 24) он должен первым ходом получить позицию (4, 25), из начальных позиций (5, 24) и (6, 23) Петя после первого хода должен получить позицию (6, 24). Позиции (4, 25) и (6, 24) рассмотрены при разборе задания 1. В этих позициях выигрышная стратегия есть у игрока, который будет ходить вторым (теперь это Петя). Эта стратегия описана при разборе задания 1. Таким образом, Петя при любой игре Вани выигрывает своим вторым ходом.

Задание 3. В начальной позиции (5, 23) выигрышная стратегия есть у Вани. После первого хода Пети может возникнуть одна из четырёх позиций: (6, 23), (5, 24), (10, 23) и (5, 46). В позициях (10, 23) и (5, 46) Ваня может выиграть одним ходом, удвоив количество камней во второй куче. Позиции (6, 23) и (5, 24) были рассмотрены при разборе задания 2. В этих позициях у игрока, который должен сделать ход (теперь это Ваня), есть выигрышная стратегия. Эта стратегия описана при разборе задания 2. Таким образом, в зависимости от игры Пети Ваня выигрывает на первом или на втором ходу.

Примечание для эксперта. Последняя фраза в приведённом решении избыточна. Не будет ошибкой, если экзаменуемый просто напишет, например, «При выбранной стратегии партия длится не более двух ходов».

В таблице изображено дерево возможных партий при описанной стратегии Вани. Заключительные позиции (в них выигрывает Ваня) выделены жирным шрифтом.

Исходное положение	Положения после очередных ходов			
	1-й ход Пети (разобраны все ходы, указана полученная позиция)	1-й ход Вани (только ход по стратегии, указана полученная позиция)	2-й ход Пети (разобраны все ходы, указана полученная позиция)	2-й ход Вани (только ход по стратегии, указана полученная позиция)
(5, 23) Всего: 28	(5, 23+1) = (5, 24) Всего: 29	(5+1, 24) = (6, 24) Всего: 30	(6+1, 24) = (7, 24) Всего: 31	(7, 24*2) = (7, 48) Всего: 55
			(6, 24+1) = (6, 25) Всего: 31	(6, 25*2) = (6, 50) Всего: 56
			(6*2, 24) = (12, 24) Всего: 36	(12, 24*2) = (12, 48) Всего: 60
			(6, 24*2) = (6, 48) Всего: 54	(6, 48*2) = (6, 96) Всего: 102
	(5+1, 23) = (6, 23) Всего: 29	(6, 23+1) = (6, 24) Всего: 30	(6+1, 24) = (7, 24) Всего: 31	(7, 24*2) = (7, 48) Всего: 55
			(6, 24+1) = (6, 25) Всего: 31	(6, 25*2) = (6, 50) Всего: 56
			(6*2, 24) = (12, 24) Всего: 36	(12, 24*2) = (12, 48) Всего: 60
			(6, 24*2) = (6, 48) Всего: 54	(6, 48*2) = (6, 96) Всего: 102
	(5*2, 23) = (10, 23) Всего: 33	(10, 23*2) = (10, 46) Всего: 56		
	(5, 23*2) = (5, 46) Всего: 51	(5, 46*2) = (5, 92) Всего: 97		

Примечание для эксперта. Дерево всех партий может быть также изображено в виде ориентированного графа – так, как показано на рисунке, или другим способом. Например, вершины дерева, соответствующие одной и той же позиции, на рисунке могут быть «склеены». Важно, чтобы множество полных путей в графе находилось во взаимно однозначном соответствии с множеством партий, возможных при описанной в решении стратегии.

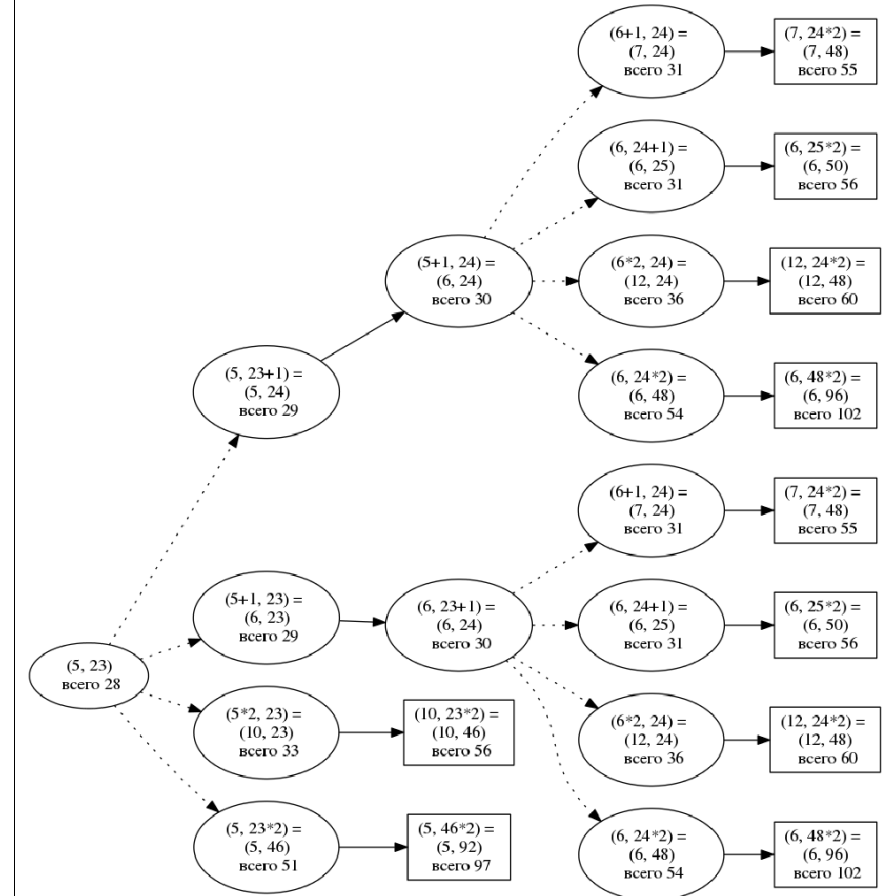


Рис. 1. Дерево всех партий, возможных при описанной стратегии Вани. Ходы Пети показаны пунктирными стрелками, ходы Вани показаны сплошными стрелками. Заключительные позиции обозначены прямоугольником

Примечание для эксперта. В некоторых позициях у Вани есть и другой

способ выигрыша: например, в позиции (6, 48) можно добавить один камень в любую кучу. То, что это не указано, не является ошибкой. Экзаменуемый не должен указывать все возможные выигрышные стратегии

Указания по оцениванию	Баллы
В задаче от ученика требуется выполнить три задания. Количество баллов в целом соответствует количеству выполненных заданий (подробнее см. ниже).	
Ошибка в решении, не искажающая основного замысла и не приведшая к неверному ответу, например арифметическая ошибка при вычислении количества камней в заключительной позиции, при оценке решения не учитывается.	
Во всех случаях стратегии могут быть описаны так, как это сделано в примере решения, или другим способом	
Выполнены все три задания.	3
Здесь и далее в решениях допускаются арифметические ошибки, которые не искажают сути решения и не приводят к неправильному ответу	
Не выполнены условия, позволяющие поставить 3 балла, и выполнено хотя бы одно из следующих условий. <ul style="list-style-type: none"> – Выполнено задание 3. – Выполнены задания 1 и 2 	2
Не выполнены условия, позволяющие поставить 2 или 3 балла, и выполнено хотя бы одно из следующих условий. <ul style="list-style-type: none"> – Выполнено задание 1. – Выполнено задание 2 	1
Не выполнено ни одно из условий, позволяющих поставить 1, 2 или 3 балла	0
<i>Максимальный балл</i>	3

РАЗДЕЛ 5

Ответом к заданиям 1–6 является одна цифра или число. Ответы к заданиям 1–6 запишите в отведённом месте работы. Для ответов к заданиям 7 и 8 используйте дополнительный лист. Ответы пишите чётко и разборчиво.

1

Программирование

Запишите число, которое будет напечатано в результате выполнения следующей программы. Для Вашего удобства программа представлена на пяти языках программирования.

Бейсик	Python
<pre>DIM S, N AS INTEGER S = 301 N = 0 WHILE S > 0 S = S - 10 N = N + 2 WEND PRINT N</pre>	<pre>s = 301 n = 0 while s > 0: s = s - 10 n = n + 2 print(n)</pre>
Алгоритмический язык	Паскаль
<pre>алг нач цел n, s s := 301 n := 0 нц пока s > 0 s := s - 10 n := n + 2 кц вывод n кон</pre>	<pre>var s, n: integer; begin s := 301; n := 0; while s > 0 do begin s := s - 10; n := n + 2; end; writeln(n) end.</pre>
Си	
<pre>#include<stdio.h> int main() { int s = 301, n = 0; while (s > 0) { s = s - 10; n = n + 2; } printf("%d\n", n); return 0; }</pre>	

Ответ: _____.

2

Запишите число, которое будет напечатано в результате выполнения следующей программы. Для Вашего удобства программа представлена на пяти языках программирования.

Бейсик	Python
<pre>DIM S, N AS INTEGER S = 0 N = 76 WHILE S < 71 S = S + 8 N = N - 3 WEND PRINT N</pre>	<pre>s = 0 n = 76 while s < 71: s = s + 8 n = n - 3 print(n)</pre>
Алгоритмический язык	Паскаль
<pre>алг нач цел n, s s := 0 n := 76 нц пока s < 71 s := s + 8 n := n - 3 кц вывод n кон</pre>	<pre>var s, n: integer; begin s := 0; n := 76; while s < 71 do begin s := s + 8; n := n - 3; end; writeln(n) end.</pre>
Си	
<pre>#include<stdio.h> int main() { int s = 0, n = 76; while (s < 71) { s = s + 8; n = n - 3; } printf("%d\n", n); return 0; }</pre>	

Ответ: _____.

3

Запишите число, которое будет напечатано в результате выполнения следующей программы. Для Вашего удобства программа представлена на пяти языках программирования.

Бейсик	Python
<pre>DIM S, N AS INTEGER S = 50 N = 1 WHILE S > 0 S = S \ 4 N = N * 3 WEND PRINT N</pre>	<pre>s = 50 n = 1 while s > 0: s = s // 4 n = n * 3 print(n)</pre>
Алгоритмический язык	Паскаль
<pre>алг нач цел s, n s := 50 n := 1 нц пока s > 0 s := div(s, 4) n := n * 3 кц вывод n кон</pre>	<pre>var s, n: integer; begin s := 50; n := 1; while s > 0 do begin s := s div 4; n := n * 3; end; writeln(n) end.</pre>
Си	
<pre>#include<stdio.h> int main() { int s = 50, n = 1; while (s > 0) { s = s / 4; n = n * 3; } printf("%d\n", n); return 0; }</pre>	

Ответ: _____.

4

Напишите в ответе число, которое будет напечатано в результате выполнения следующего алгоритма. Для Вашего удобства алгоритм представлен на пяти языках программирования.

Бейсик	Python
<pre>DIM A, B, T, M, R AS LONG 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+R FUNCTION F (x) F = 2*(x*x-36)*(x*x-36)+41 END FUNCTION</pre>	<pre>def F(x): return 2*(x*x-36)*(x*x-36)+41 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+R)</pre>
Алгоритмический язык	Паскаль
<pre>алг нач цел a, b, t, M, R a := -20; b := 20 M := a; R := F(a) нц для t от a до b если F(t) <= R то M := t; R := F(t) все кц вывод M+R кон алг цел F(цел x) нач знач := 2*(x*x-36)*(x*x-36)+41 кон</pre>	<pre>var a, b, t, M, R :longint; function F(x: longint) : longint; begin F := 2*(x*x-36)*(x*x-36)+41; 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+R) end.</pre>

Си

```
#include<stdio.h>
long F(long x)
{
    return 2*(x*x-36)*(x*x-36)+41;
}
int main()
{
    long 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);
        }
    }
    printf("%ld", M+R);
    return 0;
}
```

Ответ: _____.

5

Напишите в ответе число, равное количеству различных значений входной переменной k , при которых приведённая ниже программа выводит тот же ответ, что и при входном значении $k = 10$. Значение $k = 10$ также включается в подсчёт различных значений k . Для Вашего удобства программа приведена на пяти языках программирования.

Бейсик	Python
<pre>DIM K, I AS LONG INPUT K I = 1 WHILE F(I) < K I = I + 1 WEND IF F(I)-K <= K-F(I-1) THEN PRINT I ELSE PRINT I-1 END IF FUNCTION F(N) F = N * N * N END FUNCTION</pre>	<pre>def f(n): return n*n*n i = 1 k = int(input()) while f(i) < k: i+=1 if (f(i)-k <= k-f(i-1)): print (i) else: print (i - 1)</pre>

Алгоритмический язык	Паскаль
<pre> алг нач цел i, k ввод k i := 1 нц пока f(i) < k i := i + 1 кц если f(i)-k <= k-f(i-1) то вывод i иначе вывод i-1 все кон алг цел f(цел n) нач знач := n * n * n кон </pre>	<pre> var k, i : longint; function f(n: longint) : longint; begin f := n * n * n; end; begin readln(k); i := 1; while f(i) < k do i := i+1; if f(i)-k <= k-f(i-1) then writeln(i) else writeln(i-1); end. end. </pre>
Си	
<pre> #include<stdio.h> long f(long n) { return n * n * n; } void main() { long k, i; scanf("%ld", &k); i = 1; while (f(i)<k) i++; if (f(i)-k <= k-f(i-1)){ printf("%ld", i); } else { printf("%ld", i-1); } } </pre>	

Ответ: _____.

6

Напишите в ответе наименьшее значение входной переменной k , при котором программа выдаёт тот же ответ, что и при входном значении $k=27$. Для Вашего удобства программа приведена на пяти языках программирования.

Бейсик	Python
<pre> DIM K, I AS LONG INPUT K I = 1 WHILE F(I) < G(K) I = I + 1 WEND PRINT I FUNCTION F(N) F = N * N * N END FUNCTION FUNCTION G(N) G = 2*N + 2 END FUNCTION </pre>	<pre> def f(n): return n*n*n def g(n): return 2*n+2 k = int(input()) i = 1 while f(i) < g(k): i+=1 print (i) </pre>
Алгоритмический язык	Паскаль
<pre> алг нач цел i, k ввод k i := 1 нц пока f(i) < g(k) i := i + 1 кц вывод i кон алг цел f(цел n) нач знач := n * n * n кон алг цел g(цел n) нач знач := 2*n + 2 кон </pre>	<pre> var k, i : longint; function f(n: longint): longint; begin f := n * n * n; end; function g(n: longint): longint; begin g := 2*n + 2; end; begin readln(k); i := 1; while f(i) < g(k) do i := i+1; writeln(i) end. </pre>

Си

```
#include<stdio.h>
long f(long n) {
    return n * n * n;
}

long g(long n) {
    return 2*n + 2;
}

int main()
{
    long k, i;
    scanf("%ld", &k);
    i = 1;
    while(f(i)<g(k))
        i++;
    printf("%ld", i);
    return 0;
}
```

Ответ: _____.

7

На обработку поступает последовательность из четырёх неотрицательных целых чисел (некоторые числа могут быть одинаковыми). Нужно написать программу, которая выводит на экран количество кратных 11 чисел в исходной последовательности и минимальное кратное 11 число. Если чисел, кратных 11, нет, требуется на экран вывести «NO». Известно, что вводимые числа не превышают по абсолютной величине 1000. Программист написал программу неправильно. Ниже эта программа для Вашего удобства приведена на пяти языках программирования.

Напоминание: 0 делится на любое натуральное число.

Бейсик	Python
<pre>CONST n = 4 count = 0 minimum = 0 FOR I = 1 TO n INPUT x IF x mod 11 = 0 THEN count = count + 1 IF x > minimum THEN minimum = x END IF END IF NEXT I IF count > 0 THEN PRINT count PRINT minimum ELSE PRINT "NO" END IF</pre>	<pre>n = 4 count = 0 minimum = 0 for i in range(1, n+1): x = int(input()) if x % 11 == 0: count += 1 if x > minimum: minimum = x if count > 0: print(count) print(minimum) else: print("NO")</pre>

Алгоритмический язык

```
алг
нач
    цел n = 4
    цел i, x, minimum, count
    count := 0
    minimum := 0
    нц для i от 1 до n
        ввод x
        если mod(x, 11) = 0 то
            count := count + 1
            если x > minimum то
                minimum := x
            все
        все
    кц
    если count > 0 то
        вывод count, нс
        вывод minimum
    иначе
        вывод "NO"
    все
кон
```

Паскаль

```
const n = 4;
var i, x, minimum, count: integer;
begin
    count := 0;
    minimum := 0;
    for i := 1 to n do
        begin
            read(x);
            if x mod 11 = 0 then
                begin
                    count := count + 1;
                    if x > minimum then
                        minimum := x
                    end
                end
            end;
        if count > 0 then
            begin
                writeln(count);
                writeln(minimum);
            end
        else
            writeln('NO')
        end.
end.
```

Си

```
#include <stdio.h>
#define n 4

int main()
{
    int i, x, minimum, count;
    count = 0;
    minimum = 0;
    for (i = 1; i <= n; i++)
    {
        scanf("%d", &x);
        if (x % 11 == 0)
        {
            count++;
            if (x > minimum)
                minimum = x;
        }
    }
    if (count > 0)
    {
        printf("%d\n", count);
        printf("%d\n", minimum);
    }
    else
        printf("NO\n");
    return 0;
}
```

Последовательно выполните следующее.

1. Напишите, что выведет эта программа при вводе последовательности: 33, 34, 44, 45.

2. Приведите пример последовательности, в которой есть хотя бы одно кратное 11 число, при вводе которой, несмотря на ошибки, программа печатает 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.

Достаточно указать ошибки и способ их исправления для одного языка программирования.

Обратите внимание, что требуется найти ошибки в имеющейся программе, а не написать свою, возможно, использующую другой алгоритм решения. Исправление ошибки должно затрагивать только строку, в которой находится ошибка.

8

В физической лаборатории проводится долговременный эксперимент по изучению гравитационного поля Земли. По каналу связи каждую минуту передаются положительные целые числа – текущие показания прибора «Сигма-2015». После того как передана серия измерений (количество измерений в серии заранее известно), прибор передаёт контрольное значение серии – наибольшее число R , удовлетворяющее следующим условиям:

1) R – произведение двух различных переданных элементов последовательности («различные» означает, что не рассматриваются квадраты переданных чисел, произведения различных, но равных по величине элементов допускаются);

2) R делится на 33.

В результате помех при передаче как сами числа, так и контрольное значение могут быть искажены.

Напишите эффективную по времени и используемой памяти программу (укажите используемую версию языка программирования, например Free Pascal 2.6.4), которая будет проверять правильность контрольного значения.

Программа считается эффективной по времени, если время работы программы пропорционально количеству полученных показаний прибора N , т.е. при увеличении N в k раз время работы программы должно увеличиваться не более чем в k раз.

Программа считается эффективной по памяти, если размер памяти, использованной в программе для хранения данных, не зависит от числа N и не превышает 1 килобайта.

Программа должна напечатать отчёт по следующей форме.

Вычисленное контрольное значение: ...

Контроль пройден (или Контроль не пройден)

Если удовлетворяющее условию контрольное значение определить невозможно, то выводится только фраза «Контроль не пройден».

Перед текстом программы кратко опишите используемый Вами алгоритм решения.

На вход программе в первой строке подаётся количество чисел $N \leq 100\,000$. В каждой из последующих N строк записано одно положительное целое число, не превышающее 1000. В последней строке записано контрольное значение.

Пример входных данных:

5

66

18

44

70

29

4620

Пример выходных данных для приведённого выше примера входных данных:

Вычисленное контрольное значение: 4620

Контроль пройден

Ответы к заданиям

№ задания	Ответ
1	62
2	49
3	27
4	47
5	13
6	13

Критерии оценивания заданий с развёрнутым ответом

7

На обработку поступает последовательность из четырёх неотрицательных целых чисел (некоторые числа могут быть одинаковыми). Нужно написать программу, которая выводит на экран количество кратных 11 чисел в исходной последовательности и минимальное кратное 11 число. Если чисел, кратных 11, нет, требуется на экран вывести «NO». Известно, что вводимые числа не превышают по абсолютной величине 1000. Программист написал программу неправильно. Ниже эта программа для Вашего удобства приведена на пяти языках программирования.

Напоминание: 0 делится на любое натуральное число.

Бейсик	Python
<pre>CONST n = 4 count = 0 minimum = 0 FOR I = 1 TO n INPUT x IF x mod 11 = 0 THEN count = count + 1 IF x > minimum THEN minimum = x END IF END IF NEXT I IF count > 0 THEN PRINT count PRINT minimum ELSE PRINT "NO" END IF</pre>	<pre>n = 4 count = 0 minimum = 0 for i in range(1, n+1): x = int(input()) if x % 11 == 0: count += 1 if x > minimum: minimum = x if count > 0: print(count) print(minimum) else: print("NO")</pre>

Алгоритмический язык	Паскаль
<pre>алг нач цел n = 4 цел i, x, minimum, count count := 0 minimum := 0 нц для i от 1 до n ввод x если mod(x, 11) = 0 то count := count + 1 если x > minimum то minimum := x все все кц если count > 0 то вывод count, нс вывод minimum иначе вывод "NO" все кон</pre>	<pre>const n = 4; var i, x, minimum, count: integer; begin count := 0; minimum := 0; for i := 1 to n do begin read(x); if x mod 11 = 0 then begin count := count + 1; if x > minimum then minimum := x end end; end; if count > 0 then begin writeln(count); writeln(minimum); end else writeln('NO') end.</pre>
<pre>Си #include <stdio.h> #define n 4 int main() { int i, x, minimum, count; count = 0; minimum = 0; for (i = 1; i <= n; i++) { scanf("%d", &x); if (x % 11 == 0) { count++; if (x > minimum) minimum = x; } } if (count > 0) { printf("%d\n", count); printf("%d\n", minimum); } else printf("NO\n"); return 0; }</pre>	

Последовательно выполните следующее.

1. Напишите, что выведет эта программа при вводе последовательности: 33, 34, 44, 45.

2. Приведите пример последовательности, в которой есть хотя бы одно кратное 11 число, при вводе которой, несмотря на ошибки, программа печатает 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.

Достаточно указать ошибки и способ их исправления для одного языка программирования.

Обратите внимание, что требуется найти ошибки в имеющейся программе, а не написать свою, возможно, использующую другой алгоритм решения. Исправление ошибки должно затрагивать только строку, в которой находится ошибка.

Содержание верного ответа и указания по оцениванию (допускаются иные формулировки ответа, не искажающие его смысла)
<p>Решение использует запись программы на Паскале. Допускается использование записи программы на любом из четырёх других языков программирования.</p> <p>1. Программа выведет два числа: 2 и 44.</p> <p>2. Пример последовательности, содержащей кратные 11 числа, для которой программа выдаёт правильный ответ: 51 53 55 57.</p> <p><i>Замечание для проверяющего. В конце работы программы значение переменной <code>minimum</code> всегда равно максимальному кратному 11 числу или 0, если в последовательности нет чисел, кратных 11. Соответственно, программа будет работать верно, если в последовательности максимальное кратное 11 число равно минимальному. Выведенное количество кратных 11 чисел будет правильным в любом случае.</i></p> <p>3. В программе есть две ошибки.</p> <p>Первая ошибка: неверная инициализация ответа (переменная <code>minimum</code>).</p> <p>Строка с ошибкой: <code>minimum := 0;</code></p> <p>Верное исправление: <code>minimum := 1001;</code></p> <p>Вместо 1001 может быть любое целое число, большее 990, либо <code>MAXINT</code>. Можно использовать и число 990, так как при выводе мы проверяем, есть ли в последовательности хотя бы одно кратное 11 число.</p> <p>Вторая ошибка: неверное условие перевычисления минимума.</p> <p>Строка с ошибкой: <code>if x > minimum then</code></p> <p>Верное исправление: <code>if x < minimum then</code></p>

Указания по оцениванию	Баллы
<p>Обратите внимание! В задаче требовалось выполнить четыре действия:</p> <ol style="list-style-type: none"> 1) указать, что выведет программа при конкретной входной последовательности; 2) указать пример входной последовательности, при вводе которой программа выдаёт правильный ответ; 3) исправить первую ошибку; 4) исправить вторую ошибку. <p>Для проверки правильности выполнения п. 2) нужно формально выполнить исходную (ошибочную) программу с входными данными, которые указал экзаменуемый, и убедиться в том, что результат, выданный программой, будет таким же, как и для правильной программы.</p> <p>Для действий 3) и 4) ошибка считается исправленной, если выполнены оба следующих условия:</p> <ol style="list-style-type: none"> а) правильно указана строка с ошибкой; б) указан такой новый вариант строки, что при исправлении другой ошибки получается правильная программа 	
Выполнены все четыре необходимых действия, и ни одна верная строка не указана в качестве ошибочной	3
Не выполнены условия, позволяющие поставить 3 балла. Имеет место одна из следующих ситуаций:	2
<ol style="list-style-type: none"> а) выполнены три из четырёх необходимых действий. Ни одна верная строка не указана в качестве ошибочной; б) выполнены все четыре необходимых действия. Указано в качестве ошибочной не более одной верной строки 	
Не выполнены условия, позволяющие поставить 2 или 3 балла. Выполнены два необходимых действия из четырёх	1
Не выполнены условия, позволяющие поставить 1, 2 или 3 балла	0
<i>Максимальный балл</i>	3

В физической лаборатории проводится долговременный эксперимент по изучению гравитационного поля Земли. По каналу связи каждую минуту передаются положительные целые числа – текущие показания прибора «Сигма 2015». После того как передана серия измерений (количество измерений в серии заранее известно), прибор передаёт контрольное значение серии – наибольшее число R , удовлетворяющее следующим условиям:

- 1) R – произведение двух различных переданных элементов последовательности («различные» означает, что не рассматриваются квадраты переданных чисел, произведения различных, но равных по величине элементов допускаются);
- 2) R делится на 33.

В результате помех при передаче как сами числа, так и контрольное значение могут быть искажены.

Напишите эффективную по времени и используемой памяти программу (укажите используемую версию языка программирования, например Free Pascal 2.6.4), которая будет проверять правильность контрольного значения. Программа считается эффективной по времени, если время работы программы пропорционально количеству полученных показаний прибора N , т.е. при увеличении N в k раз время работы программы должно увеличиваться не более чем в k раз.

Программа считается эффективной по памяти, если размер памяти, использованной в программе для хранения данных, не зависит от числа N и не превышает 1 килобайта.

Программа должна напечатать отчёт по следующей форме.

Вычисленное контрольное значение: ...

Контроль пройден (или Контроль не пройден)

Если удовлетворяющее условию контрольное значение определить невозможно, то выводится только фраза «Контроль не пройден».

Перед текстом программы кратко опишите используемый Вами алгоритм решения.

На вход программе в первой строке подаётся количество чисел $N \leq 100\,000$. В каждой из последующих N строк записано одно положительное целое число, не превышающее 1000. В последней строке записано контрольное значение.

Пример входных данных:

5
66
18
44
70
29
4620

Пример выходных данных для приведённого выше примера входных данных:

Вычисленное контрольное значение: 4620

Контроль пройден

Содержание верного ответа и указания по оцениванию

(допускаются иные формулировки ответа, не искажающие его смысла)

Произведение двух чисел делится на 33, если:

- один из сомножителей делится на 33 (второй может быть любым) либо
- ни один из сомножителей не делится на 33, но один из сомножителей делится на 3, а другой – на 11.

Программа, вычисляющая контрольное значение, читает все входные данные один раз, не запоминая их в массиве. Для прочитанного фрагмента входной последовательности программа хранит значения четырёх величин:

$M3$ – самое большое число, кратное 3;

$M11$ – самое большое число, кратное 11, но не кратное 3;

$M33$ – самое большое число, кратное 33;

MAX – самое большое число среди всех элементов последовательности, отличное от $M33$ (если число $M33$ встретилось более одного раза и оно является максимальным, то $MAX = M33$).

После того как все данные прочитаны, искомое контрольное значение вычисляется как максимум из произведений $M33*MAX$ и $M3*M11$.

Возможны модификации этого алгоритма. Ниже приведены программы на языках Паскаль и Си, а также на алгоритмическом языке, реализующие одну из таких модификаций. Допускаются решения, записанные на других языках программирования.

Пример правильной и эффективной программы на языке Паскаль

```
var M3,M11,M33,R,MAX,dat,res,i,N: longint;
begin
  M3 := 0;
  M11 := 0;
  M33 := 0;
  MAX := 0;
  readln(N);
  for i := 1 to N do
  begin
    readln(dat);
    if (dat mod 3 = 0) and (dat > M3) then
      M3 := dat
    else if (dat mod 11 = 0) and (dat > M11) then
      M11 := dat;
    if (dat mod 33 = 0) and (dat > M33) then
      begin
        if M33 > MAX then MAX := M33;
        M33 := dat
      end
    else if dat > MAX then MAX := dat;
  end;
  readln(R);
  if (M3*M11 < M33*MAX) then
    res := M33*MAX
  else
    res := M3*M11;
  if res > 0 then
    writeln('Вычисленное контрольное значение: ',res);
  if (R > 0) and (R = res)
    then writeln('Контроль пройден')
    else writeln('Контроль не пройден');
end.
```

Пример правильной и эффективной программы на языке Си

```
#include <stdio.h>
void main ()
{
  int N; /*количество чисел на входе*/
  int x; /*исходные данные*/
  int m3=0; /*макс. число, кратное 3*/
  int m11=0; /*макс. число, кратное 11, но не 3*/
  int m33=0; /*макс. число, кратное 33 */
  int max=0;
  int R; /*введённое контрольное значение*/
  int m; /*вычисленное контрольное значение*/
  int i;
  scanf("%d", &N);
  for (i=1; i<=N; ++i) {
    scanf("%d", &x);
    if (x % 3 == 0 && x > m3) m3 = x;
    else if (x % 11 == 0 && x > m11) m11 = x;
    if (x % 33 == 0 && x > m33) {
      if (m33 > max) max = m33;
      m33 = x;
    }
    else if (x > max) max = x;
  }
  scanf("%d", &R);
  if (m3*m11 < m33*max) m = m33*max;
  else m = m3*m11;
  if (m>0) printf("Вычисленное контрольное значение: %d\n", m);
  if (R>0 && R==m) printf("Контроль пройден\n");
  else printf("Контроль не пройден\n");
}
```

Пример правильной и эффективной программы на алгоритмическом языке

```

алг
нач
  цел N | количество чисел на входе
  цел x | исходные данные
  цел m3=0 | макс. число, кратное 3
  цел m11=0 | макс. число, кратное 11, но не 3
  цел m33=0 | макс. число, кратное 33
  цел mx=0
  цел R | введённое контрольное значение
  цел m | вычисленное контрольное значение
ввод N
нц N раз
  ввод x
  если mod(x,3) = 0 и x > m3
    то m3 := x
  иначе
    если mod(x,11) = 0 и x > m11
      то m11 := x
    все
  все
  если mod(x,33) = 0 и x > m33
    то
      если m33 > mx то mx := m33 все
      m33 := x
    иначе
      если x > mx то mx := x все
  все
кц
ввод R
если m3*m11 < m33*mx
  то m := m33*mx
  иначе m := m3*m11
все
если m>0
  то вывод нс, "Вычисленное контрольное значение: ", m
все
если R>0 и R=m
  то вывод нс, "Контроль пройден"
  иначе вывод нс, "Контроль не пройден"
все
кон

```

Указания по оцениванию	Баллы
<p>4 балла ставится за эффективную и правильно работающую программу, которая, возможно, содержит до трёх синтаксических ошибок (описок).</p> <p>3 балла ставится в случае, когда фактически задача решена, 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 (список допустимых ошибок приведён ниже).</p> <p>2 балла ставится, если в дополнение к неточностям, которые перечислены выше, программа работает неэффективно по времени и/или допущено до трёх упомянутых выше содержательных ошибок. Допускается до семи описок.</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 сказанное уточнено.</p>	
<p>Программа правильно работает для любых входных данных произвольного размера. Используемая память не зависит от количества прочитанных чисел, а время работы пропорционально этому количеству.</p> <p>Допускается наличие в тексте программы до трёх синтаксических ошибок одного из следующих видов:</p> <ol style="list-style-type: none"> 1) пропущен или неверно указан знак пунктуации; 2) неверно написано или пропущено зарезервированное слово языка программирования; 3) не описана или неверно описана переменная; 4) применяется операция, недопустимая для соответствующего типа данных. <p>Если одна и та же ошибка встречается несколько раз, это считается за одну ошибку</p>	4

<p>Не выполнены условия, позволяющие поставить 4 балла.</p> <p>Программа в целом работает правильно для любых входных данных произвольного размера. Время работы пропорционально количеству введенных чисел; правильно указано, какие величины должны вычисляться по ходу чтения элементов последовательности чисел.</p> <p>Количество синтаксических ошибок (описок) указанных выше видов – не более пяти.</p> <p>Используемая память, возможно, зависит от количества прочитанных чисел (например, входные данные запоминаются в массиве или другой структуре данных (контейнер <code>priority_queue</code>, <code>vector</code>, <code>set</code> или <code>map</code> в C++)).</p> <p>Допускается ошибка при вводе данных, неверный или неполный вывод результатов или неверная работа программы в «экзотических» ситуациях. Например, при использовании 16-битного целого (<code>integer</code> в WPascal или Qbasic) умножаются два числа этого типа (результат по условию может не помещаться в 16 бит).</p> <p>Кроме того, допускается наличие одной ошибки, принадлежащей к одному из следующих видов ошибок:</p> <ol style="list-style-type: none"> 1) ошибка при инициализации максимумов; 2) неверно обрабатывается ситуация, когда один или несколько максимумов не определены; 3) неверно обрабатывается ситуация, когда максимальное произведение получается умножением одинаковых чисел (но разных элементов массива); 4) при вычислении максимумов учитываются произведения вида <code>a[i]*a[i]</code>; 5) допущен выход за границу массива; 6) используется знак <code><</code> вместо <code><=</code>, <code>or</code> вместо <code>and</code> и т.п. 	3
---	---

<p>Не выполнены условия, позволяющие поставить 3 или 4 балла.</p> <p>Программа работает в целом верно, эффективно или нет, но в реализации алгоритма есть до трёх содержательных ошибок, допустимые виды ошибок перечислены в критериях на 3 балла.</p> <p>Количество синтаксических описок не должно быть более семи.</p> <p>Программа может быть неэффективна по времени. Например, все числа запоминаются в массиве и перебираются все возможные пары элементов последовательности:</p> <pre> max := 0; for i := 1 to N - 1 do begin for j := i + 1 to N do begin if ((a[i]*a[j]) mod 33 = 0) and (a[i]*a[j] > max) then max := a[i]*a[j]; end end; </pre>	2
<p>Не выполнены условия, позволяющие поставить 2, 3 или 4 балла.</p> <p>Программа работает правильно в отдельных частных случаях. Из описания алгоритма и общей структуры программы видно, что экзаменуемый в целом правильно представляет путь решения задачи. Допускается любое количество описок</p>	1
<p>Не выполнены критерии, позволяющие поставить 1, 2, 3 или 4 балла</p>	0
<i>Максимальный балл</i>	4