

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

№ задания	Ответ
1	7
2	cab
3	17
4	4
5	20
6	184
7	2
8	60
9	5
10	61
11	56
12	248
13	390
14	88
15	24
16	17
17	280
18	21
19	89
20	117
21	68
22	38
23	61

**Часть 2**

24

Требовалось написать программу, при выполнении которой с клавиатуры считывается натуральное число N, не превосходящее  $10^9$ , и выводится сумма цифр этого числа. Программист написал программу неправильно.

СИ	Python	Паскаль
#include <stdio.h> int main() { int N, sum, d; scanf("%d", &N);	N = int(input()) sum = 1 while N > 0: d = N % 10 N = N // 10	var N: longint; sum, d: integer; begin readln(N); sum := 1;

sum = 1; while ( N > 0 ) { d = N % 10; N = N / 10; sum = sum + 1; } printf("%d", sum); return 0; }	sum = sum + 1 print(sum)	while N > 0 do begin d := N mod 10; N := N div 10; sum := sum + 1; end; writeln(sum); end.
--	-----------------------------	--

Последовательно выполните следующее.

- Напишите, что выведет эта программа при вводе числа 256.
- Приведите пример такого трёхзначного числа X, при вводе которого программа выведет правильный ответ.
- 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:
  - выпишите строку, в которой сделана ошибка;
  - укажите, как исправить ошибку, т.е. приведите правильный вариант строки.

**Содержание верного ответа и указания по оцениванию**  
 (допускаются иные формулировки ответа, не искажающие его смысла)
 

- При вводе числа 256 программа выведет число 4.
- Программа выводит правильный ответ, если сумма цифр на 1 больше, чем количество цифр, например, для числа 400.
- В программе есть две ошибки.

1) Неверная инициализация. Стока с ошибкой:

```
sum := 1;
```

Верное исправление:

```
sum := 0;
```

2) Неверное изменение переменной sum. Стока с ошибкой:

```
sum := sum + 1;
```

Верное исправление:

```
sum := sum + d;
```

**Указания по оцениванию**

В задаче требуется выполнить **три** действия.

1. Указать результат программы при данном вводе.

**Баллы**

<p>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.</p>	
<p>Это действие считается выполненным, если указан пример числа, при вводе которого выводится верное сообщение (верный показатель степени или текст «Не существует», если введённое число не является степенью). Ученик не обязан указывать, что будет выведено, и объяснять, как работает программа.</p>	
<p>3. Найти и исправить ошибки в программе.</p>	
<p>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> <ul style="list-style-type: none"> <li>а) правильно указана строка с ошибкой;</li> <li>б) указан такой новый вариант строки, что при исправлении другой ошибки получается правильная программа</li> </ul>	
<p>Выполнены все три необходимых действия, и ни одна верная</p>	3
<p>Не выполнены условия, позволяющие поставить 3 балла. Имеет место одна из следующих ситуаций.</p>	2
<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>	
<p>Не выполнены условия, позволяющие поставить 2 или 3 балла. При этом имеет место один из следующих случаев.</p>	1
<p>1. Выполнены два первых действия. При этом несущественно, 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>	
<p>Не выполнены условия, позволяющие поставить 1, 2 или 3 балла</p>	0
<i>Максимальный балл</i>	3

25

Дан массив, содержащий 2014 положительных целых чисел. Симметричной парой называются два элемента, которые находятся на равном расстоянии от концов массива. Например, 1-й и 2014-й элементы, 2-й и 2013-й и т. д. Порядок элементов в симметричной паре не учитывается: элементы на 1 и 2014 местах – это та же самая пара, что и элементы на 2014 и 1 местах. Напишите на одном из языков программирования программу, которая подсчитывает в массиве количество симметричных пар, у которых сумма элементов больше 20. Программа должна вывести одно число – количество отобранных симметричных пар.

СИ	Python	Паскаль
#include <stdio.h> #define n 2014 int main() { int a[n]; int i, j, k; for (i = 0; i < n; i++) scanf("%d", &a[i]); ... return 0; }	# допускается также # использовать две # целочисленные # переменные j, k a = [] n = 2014 for i in range(n): a.append(int(input())) ...	const n = 2014; var a: array [0..n-1] of integer; i, j, k: integer; begin for i := 0 to n-1 do readln(a[i]); ... end.

Содержание верного ответа и указания по оцениванию (допускаются иные формулировки ответа, не искажающие его смысла)			
Паскаль	Python	Си	
k := 0;	k = 0	k = 0;	

for i:=0 to n div 2 - 1 do if a[i]+a[n- 1-i] > 20: then k:= k + 1; writeln(k);	for i in range(n // 2): if a[i]+a[n- 1-i]>20: k += 1 print(k)	for(i=0;i<n/2;i++) { if (a[i]+a[n-1-i]>20) k++; printf("%d", k);	
---	---	--	--

Указания по оцениванию	Баллы
------------------------	-------

*Общие указания*

1. В алгоритме, записанном на языке программирования, допускается наличие отдельных синтаксических ошибок, не искажающих замысла автора программы.
2. Эффективность алгоритма не имеет значения и не оценивается.
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

Предложен правильный алгоритм, выдающий в качестве результата верное значение	2
---	---

Не выполнены условия, позволяющие поставить 2 балла. Предложено в целом верное решение, содержащее не более одной ошибки из числа следующих:	1
--	---

- 1) в цикле происходит выход за границу массива (например, при использовании цикла от 1 до N);
- 2) не инициализируется или неверно инициализируется счётчик количества найденных пар;
- 3) счётчик количества пар в цикле не изменяется или изменяется неверно;

4) неверно выделяется последняя цифра числа;	
5) при проверке выполнения условия для пары элементов используются неверные индексы;	
6) последняя цифра выделяется не у самих элементов массива, а у их индексов;	
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 while) или меняется неверно;	
12) неверно расставлены операторные скобки	
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 два камня или увеличить количество камней в куче в три раза и затем добавить в кучу два камня. Игра завершается в тот момент, когда количество камней в куче становится не менее 60. Победителем считается игрок, сделавший последний ход, то есть первым получивший кучу, в которой будет 60 или больше камней. В начальный момент в куче было S камней,  $1 \leq S \leq 59$ .

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 выигравший ход для каждого указанного значения S.

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

Задание 2. Укажите все значения S, при которых у Паши есть выигрышная стратегия, причём Паши 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).

<b>Содержание верного ответа и указания по оцениванию</b> (допускаются иные формулировки ответа, не искажающие его смысла)	
<b>Указания по оцениванию</b>	<b>Баллы</b>
<p><b>Предварительные замечания</b> В задаче от ученика требуется выполнить три задания. Их трудность возрастает. 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. Пункт 1а считается выполненным, если правильно указаны все позиции, в которых Паша выигрывает первым ходом, и указано, каким должен быть первый ход. Пункт 1б считается выполненным, если (i) правильно указано, кто из игроков имеет выигрышную стратегию в каждой из указанных позиций, и (ii) описаны выигрышные стратегии – так, как это сделано в образце решения, или другим способом. Первое задание считается выполненным полностью, если выполнены полностью оба пункта: 1а и 1б. <b>Замечание для проверяющего.</b> 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,</p>	

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 (Вали), нужно перебрать все возможные первые ходы Паши и убедиться, что для всех полученных позиций мы знаем выигрышную стратегию для игрока, который ходит первым.

В примере решения мы используем в основном второй способ описания стратегии. 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 указано, кто из игроков имеет выигрышную стратегию в каждой из указанных позиций, и (ii) описаны выигрышные стратегии.**

**Задание 3 считается выполненным, если (i) правильно указано, что выигрышную стратегию имеет Валя; (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 сделано в примере решения, или другим способом**

Выполнены второе и третье задания.	3
Для первого задания правильно перечислены позиции, в которых Паша выигрывает первым ходом (п. 1а), и правильно указано, кто из игроков имеет выигрышную стратегию при указанных значениях $S$ (п. 1б). При этом допускаются недочёты следующих типов: - в п. 1а не указано, каким ходом выигрывает Паша;	

- в п. 1б не указано, что игрокам нет смысла удваивать количество камней в куче. 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 одно из следующих условий.	2
<ul style="list-style-type: none"> <li>Выполнено третье задание.</li> <li>Выполнены первое и второе задания.</li> <li>Первое задание выполнено, возможно, при наличии недочётов, указанных в критериях на 3 балла; для второго задания (i) правильно указано, кто из игроков имеет выигрышную стратегию в каждой из указанных позиций, и (ii) правильно указан первый ход Паши при выигрышной стратегии, однако не указано, что после выбранного хода Паши получается позиция, выигрышная для Вали; для третьего задания правильно указан игрок, имеющий выигрышную стратегию</li> </ul>	
Не выполнены условия, позволяющие поставить 3 или 2 балла, и выполнено одно из следующих условий.	1
<ul style="list-style-type: none"> <li>Первое задание выполнено, возможно, с недочётами, указанными в критериях на 3 балла.</li> <li>Второе задание выполнено, возможно, с недочётами, указанными в критериях на 2 балла.</li> <li>Для второго и третьего заданий во всех случаях правильно указан игрок, имеющий выигрышную стратегию</li> </ul>	
Не выполнено ни одно из условий, позволяющих поставить 1, 2 или 3 балла	0
<i>Максимальный балл</i>	3

27

Гоночная Дед Мороз и Снегурочка приходят на детские утренники с мешком конфет. Дед Мороз делит конфеты поровну между всеми присутствующими детьми (детей на утреннике никогда не бывает больше 100), а оставшиеся конфеты отдает Снегурочки. Снегурочка каждый раз записывает в блокнот количество полученных конфет. Если конфеты разделились между всеми детьми без остатка, Снегурочка ничего не получает и ничего не записывает. Когда утренники закончились, Деду Морозу стало интересно, какое число чаще всего записывала Снегурочка. Дед Мороз и Снегурочка – волшебные,

поэтому число утренников  $N$ , на которых они побывали, может быть очень большим. Напишите программу, которая будет решать эту задачу.

В первой строке вводится одно целое положительное число – количество утренников  $N$ .

Каждая из следующих  $N$  строк содержит два целых числа: сначала  $D$  – количество пришедших на очередной утренник детей, а затем  $K$  – количество конфет в мешке Деда Мороза на этом утреннике. Гарантируется выполнение следующих соотношений:

$$1 \leq N \leq 10000$$

$$1 \leq D \leq 100 \text{ (для каждого } D\text{)}$$

$$D \leq K \leq 1000 \text{ (для каждой пары } D, K\text{)}$$

Программа должна вывести одно число – то, которое Снегурочка записывала чаще всего. Если несколько чисел записывались одинаково часто, надо вывести большее из них. Если Снегурочка ни разу ничего не записывала, надо вывести ноль.

Пример входных данных:

```
7
10 58
15 315
20 408
100 1000
32 63
32 63
11 121
```

Пример выходных данных для приведённого выше примера входных данных:

31

**Содержание верного ответа и указания по оцениванию**

(допускаются иные формулировки ответа, не искажающие его смысла)

```
const MAX=100;
```

```
var count: array[0..MAX-1] of integer;
```

```

i, ost, nMax, N, D, K: integer;
begin
readln(N);
for i:=1 to N do begin
  readln(D, K);
  ost := K mod D;
  count[ost] := count[ost] + 1;
end;
nMax:=1;
for i:=2 to MAX-1 do
  if count[i] >= count[nMax] then
    nMax:=i;
if count[nMax] = 0 then
  nMax := 0;
writeln(nMax);
end.

```

Указания по оцениванию	Баллы
<p><b>Предварительные замечания.</b></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. Случай 2.1. Ученик представил только одну программу.</p> <p>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.</p> <p>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.</p> <p>Следует первую (по порядку в представленных учеником материалах) программу рассматривать как ответ на задание А, вторую – как ответ на задание Б.</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.

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.

**НАПОМИНАЕМ!** Итоговый балл за задачу – это больший из баллов, полученных учеником за каждое из двух представленных решений.

#### Пояснения для проверяющих.

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.

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.

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.

3. Приведённые в п. 2.1–2.5 правила имеют целью избежать снижения баллов из-за того, что ученик перепутал обозначения заданий.

#### Общие принципы оценивания решений

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»).

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

<p>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>	
<b>Критерии оценивания задания А</b>	
<p>При решении задачи А программа верно находит требуемую сумму для любых 6 пар исходных данных.</p> <p>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 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 или 2 балла</p> <p style="text-align: right;"><i>Максимальный балл для задания А</i></p>	0
<b>Критерии оценивания задания Б</b>	2
<p>Программа правильно работает для любых соответствующих условий 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.</p> <p>Программа может содержать не более трёх синтаксических ошибок следующих видов:</p>	4
<p>1) пропущен или неверно указан знак пунктуации;</p>	

<p>2) неверно написано или пропущено зарезервированное слово языка программирования;</p> <p>3) не описана или неверно описана переменная;</p> <p>4) применяется операция, недопустимая для соответствующего типа данных.</p> <p>К синтаксическим ошибкам приравнивается использование неверного типа данных.</p> <p>Если одна и та же ошибка встречается несколько раз, она считается за одну ошибку</p>	
<p>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.</p> <p>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.</p> <p>Программа может содержать не более пяти синтаксических и приравненных к ним ошибок, описанных в критериях на 4 балла. Кроме того, допускается наличие одной ошибки, принадлежащей к одному из следующих видов:</p> <ul style="list-style-type: none"> <li>1) ошибка инициализации, в том числе отсутствие инициализации;</li> <li>2) не выводится результат, равный 0, или вместо 0 выводится неверное значение;</li> <li>3) допущен выход за границу массива;</li> <li>4) используется знак "&lt;" вместо "&lt;=", "ог" вместо "and" и т.п.</li> </ul>	3
<p>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.</p> <p>Количество синтаксических «описок» не должно быть более семи. Программа может быть неэффективна по времени,</p>	2

например все числа запоминаются в массиве и перебираются все возможные суммы, т.е., по сути, реализовано решение задачи А без ограничений на количество введённых пар	
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