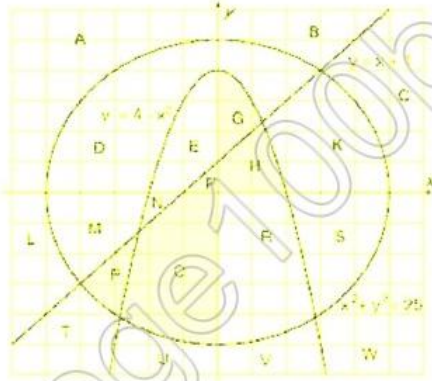


Критерии оценивания заданий с развёрнутым ответом

C1 Требовалось написать программу, при выполнении которой с клавиатуры считываются координаты точки на плоскости (x, y – действительные числа) и определяется принадлежность этой точки заданной закрашенной области (включая границы).



Ученик написал такую программу:

Бейсик

```
INPUT x, y
IF x*x + y*y <= 25 THEN
  IF y <= 4-x*x THEN
    IF y <= x+1 THEN
      PRINT "принадлежит"
    ELSE
      PRINT "не принадлежит"
    END IF
  END IF
END IF
END IF
END
```

Паскаль

```
var x, y: real;
begin
  readln(x, y);
  if x*x + y*y <= 25 then
    if y <= 4-x*x then
      if y <= x+1 then
        write('принадлежит')
      else
        write('не принадлежит')
      end if
    end if
  end if
end.
```

Cп

```
#include <stdio.h>
void main(){
  float x, y;
  scanf("%f %f",&x, &y);
  if (x*x + y*y <= 25)
    if (y <= 4-x*x)
      if (x*x + y*y <= 25)
        if (y <= x+1)
          printf("принадлежит");
        else
          printf("не принадлежит");
      }
  }
}
```

Алгоритмический язык

```
алг
нач
  вещ x, y
  ввод x, y
  если x*x + y*y <= 25 то
    если y <= 4-x*x то
      если y <= x+1 то
        вывод 'принадлежит'
      иначе
        вывод 'не принадлежит'
      все
    все
  все
кон
```

При проверке работы программа выполнялась по шагам для некоторых контрольных значений x и y , при этом был заполнен протокол тестирования, содержащий следующую информацию.

Область – часть плоскости, которой принадлежит проверяемая точка (все возможные области отмечены на рисунке буквами А, В, С, ... W).

Условие 1, Условие 2, Условие 3 – результат проверки соответствующего условия (да или нет). Если условие не проверялось, в протокол записывался прочерк.

Вывод – сообщение, которое вывела программа. Если программа ничего не вывела, в протокол записывался прочерк.

Верно – итоговое заключение (да или нет) о правильности результата работы программы при данных значениях x и y .

В результате неаккуратного обращения протокол был испорчен, частично сохранились только три строки:

Область	Условие 1 ($x*x + y*y \leq 25$)	Условие 2 ($y \leq 4 - x*x$)	Условие 3 ($y \leq x+1$)	Вывод	Верно
	да			—	
				принадлежит	нет
			нет		нет

Последовательно выполните следующее.

1. Восстановите уцелевшие строки протокола, заполнив все клетки таблицы. Там, где содержание восстанавливается неоднозначно, запишите любое возможное значение. Например, если для нескольких областей получается одинаковая строка таблицы, укажите в графе «Область» любую из этих областей.

2. Укажите, как нужно доработать программу, чтобы не было случаев её неправильной работы (это можно сделать несколькими способами, достаточно указать любой способ доработки исходной программы).

Элементы ответа:

Правильно заполненная таблица:

Область	Условие 1 ($x*x + y*y \leq 25$)	Условие 2 ($y \leq 4 - x*x$)	Условие 3 ($y \leq x+1$)	Вывод	Верно
D K M P S	да	нет	—	—	нет
R	да	да	да	принадлежит	нет
G	да	да	нет	не принадлежит	нет

Все ячейки таблицы, кроме первой ячейки первой строки, заполняются однозначно. Для этой ячейки в образце перечислены все возможные области. Таким образом, строка таблицы в работе экзаменуемого заполнена верно, если в графе «Область» указана одна из букв, приведённых в образце, а остальные графы полностью совпадают с образцом. Если у экзаменуемого в графе «Область» указано более одной буквы, заполнение считается верным, только если *все* указанные в работе буквы допустимы, то есть присутствуют в соответствующей строке образца. Например, для первой строки допустимыми записями в графе «Область» могут быть «D», «KM», «SPK» и т.д. Примеры ошибочных записей в этой клетке: «A», «DKU», «SG». Обратите внимание: если указано несколько букв, среди которых есть хотя бы одна

часть и описать каждую из них. Например, можно объединить области G, H, ограниченные параболой и осями координат, и области F, P, Q, ограниченные прямой, окружностью и осью ординат. При этом получается такой фрагмент программы (пример на Паскале):

```
if (y<=4-x*x) and (y>=0) and (x>=0) or
(y<=x+1) and (y<=0) and (x*x+y*y<=25) then
write('принадлежит')
else
write('не принадлежит')
```

Возможны и другие способы доработки программы.

Обратите внимание! В задаче требуется выполнить три действия.

1. Заполнить таблицу.
2. Исправить ошибку в условном операторе (отсутствие разбора случая ELSE).
3. Исправить ошибку, связанную с неправильным набором условий.

Баллы за данное задание начисляются как сумма баллов за верное выполнение каждого действия.

Рассмотрим отдельно каждое действие.

1. Действие по заполнению таблицы считается выполненным, если в таблице нет ошибок.

2. В исходной программе неправильно использован условный оператор, в результате чего при невыполнении первого или второго условия программа не выдаёт ничего (отсутствуют случаи ELSE). Исправлением этой ошибки может быть либо добавление случая ELSE к каждому условию IF, либо объединение всех условий IF в одно при помощи конъюнкции.

В сложных случаях это действие считается выполненным, если программа выдает одно из двух сообщений: «принадлежит» или «не принадлежит» для любых пар чисел x, y , и при этом программа не стала работать хуже, чем раньше, т.е. для всех точек, для которых программа ранее выдавала верный ответ, доработанная программа также выдает верный ответ.

3. Приведённые в исходной программе ограничения не описывают требуемые области. Исправлением этой ошибки может быть разбиение области на части и использование дизъюнкции либо использование комбинации каскадных условий.

В сложных случаях это действие считается выполненным, если верно определены заштрихованные области, т.е. программа выводит сообщение «принадлежит» для всех точек закрасшенных областей и только для них, для точек вне заштрихованных областей программа выводит «не принадлежит» или не выводит ничего.

Указания по оцениванию	Баллы
<p>Выполнены все три действия: верно заполнена таблица, исправлены две ошибки. Программа для всех пар чисел x, y верно определяет принадлежность или непринадлежность точки закрашенной области. Во фрагментах программ допускается наличие отдельных синтаксических ошибок, не искажающих замысла автора решения.</p> <p>При написании операций сравнения допускается одно неправильное использование строгих/нестрогих неравенств (считается несущественной ошибкой, погрешностью записи). Например, вместо «$y \geq 0$» используется «$y > 0$».</p>	3
<p>Правильно выполнены два действия из трёх: исправлены обе ошибки, но таблица отсутствует либо содержит ошибки, или же приведена верная таблица, но исправлена только одна ошибка программы.</p> <p>Два балла ставятся также в случае, если таблица заполнена верно, а в программе правильно записаны все условия, но логическое выражение получилось неверным из-за ошибки в учёте приоритета операций (не расставлены или неверно расставлены скобки).</p> <p>При оценивании этого задания на 2 балла допускается не учитывать корректность работы программ на точках границ областей (вместо нестрогих в решении были использованы строгие неравенства).</p>	2
<p>Правильно выполнено только одно действие из трёх, то есть не выполнены условия, позволяющие поставить 2 или 3 балла, и имеет место одна из следующих ситуаций.</p> <p>1. Таблица заполнена и содержит ошибки не более чем в одной строке; ни одна из ошибок не исправлена.</p> <p>2. Таблица не заполнена (или заполнена и содержит ошибки более чем в одной строке); исправлена ровно одна ошибка программы. При этом допускается не учитывать корректность работы программ на точках границ областей (вместо нестрогих неравенств в решении были использованы строгие).</p>	1
<p>Не выполнены условия, позволяющие поставить 1, 2 или 3 балла, то есть таблица анализа правильности алгоритма не заполнена либо содержит ошибки в двух и более строках; программа не приведена либо приведена, но ни одна из двух ошибок не исправлена).</p>	0
<i>Максимальный балл</i>	3

C2 Дан массив, содержащий 70 целых чисел. Опишите на одном из языков программирования *эффективный* алгоритм, позволяющий найти и вывести наибольшую разность двух чисел, содержащихся в массиве. Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать часть из описанных переменных.

Паскаль	<pre>const N=70; var a: array [1..N] of integer; i, j, x, y: integer; begin for i:=1 to N do readln(a[i]); ... end.</pre>
Бейсик	<pre>N=70 DIM A(N) AS INTEGER DIM I, J, X, Y AS INTEGER FOR I = 1 TO N INPUT A(I) NEXT I ... END</pre>
Си	<pre>#include <stdio.h> #define N 70 void main(){ int a[N]; int i, j, x, y; for (i=0; i<N; i++) scanf("%d", &a[i]); ... }</pre>
Алгоритмический язык	<pre>алг нач цел N=70 целтаб a[1:N] цел i, j, x, y нц для i от 1 до N ввод a[i] кц ... кон</pre>

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

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

Максимальная разность двух элементов достигается, когда один из них максимален, а второй минимален. Таким образом, задача сводится к нахождению в массиве минимального и максимального элементов и вычислению их разности.

Пример программы на Паскале

```
x:=a[1]; y:=a[1];
for i:=2 to N do begin
    if a[i]>x then x:=a[i];
    if a[i]<y then y:=a[i];
end;
writeln(x-y);
```

Указания по оцениванию	Баллы
Предложен правильный алгоритм, решающий задачу за один проход по массиву и выдающий верное значение. Допускается запись алгоритма на другом языке, использующая аналогичные переменные. В случае если язык программирования использует типизированные переменные, описания переменных должны быть аналогичны описаниям переменных на языках, использованных в задании. 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. В алгоритме, записанном на языке программирования, допускается наличие отдельных синтаксических ошибок, не искажающих замысла автора программы.	2
Предложено верное, но недостаточно эффективное решение, например вычисляется разность каждой пары элементов массива и из полученных значений выбирается наибольшее. В любом (эффективном или нет) варианте решения может присутствовать не более одной ошибки из числа следующих. 1. Не инициализируются или неверно инициализируются переменные для хранения минимального и/или максимального значения. 2. Вместо значения элемента проверяется его индекс. 3. При вычислении разности из минимума вычитается максимум. 4. Отсутствует вывод ответа. 5. Используется переменная, не объявленная в разделе описания переменных. 6. Не указано или неверно указано условие завершения цикла. 7. Индексная переменная в цикле не меняется (например, в цикле while) или меняется неверно.	1
Ошибок, перечисленных выше, две или больше, или алгоритм сформулирован неверно.	0
<i>Максимальный балл</i>	2

С3 Два игрока, Петя и Ваня, играют в следующую игру. Перед игроками лежит куча камней. Игроки ходят по очереди, первый ход делает Петя. За один ход игрок может добавить в кучу один камень или увеличить количество камней в куче в пять раз. Например, имея кучу из 10 камней, за один ход можно получить кучу из 11 или 50 камней. У каждого игрока, чтобы делать ходы, есть неограниченное количество камней.

Игра завершается в тот момент, когда количество камней в куче становится более 100. Победителем считается игрок, сделавший последний ход, то есть первым получивший кучу, в которой будет 101 или больше камней.

В начальный момент в куче было S камней, $1 \leq S \leq 100$.

Говорят, 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.

Выполните следующие задания. Во всех случаях обосновывайте свой ответ.

1. а) При каких значениях числа S Петя может выиграть первым ходом? Укажите все такие значения и выигрывающий ход Пети.

б) Укажите такое значение S , при котором Петя не может выиграть за один ход, но при любом ходе Пети Ваня может выиграть своим первым ходом. Опишите выигрышную стратегию Вани.

2. Укажите два значения S , при которых у Пети есть выигрышная стратегия, причём Петя не может выиграть первым ходом, но Петя может выиграть своим вторым ходом независимо от того, как будет ходить Ваня.

Для указанных значений S опишите выигрышную стратегию Пети.

3. Укажите такое значение S , при котором
– у Вани есть выигрышная стратегия, позволяющая ему выиграть первым или вторым ходом при любой игре Пети, и при этом

– у Вани нет стратегии, которая позволит ему гарантированно выиграть первым ходом.

Для указанного значения S опишите выигрышную стратегию Вани. Постройте дерево всех партий, возможных при этой выигрышной стратегии Вани. Представьте его в виде рисунка или таблицы. Для каждого ребра дерева укажите, кто делает ход, для каждого узла – количество камней в позиции.

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

1. а) Петя может выиграть, если $S = 21, \dots, 100$. Пете достаточно увеличить количество камней в 5 раз. При $S < 21$ получить за один ход больше 100 камней невозможно.

б) Ваня может выиграть первым ходом (как бы ни играл Петя), если исходно в куче будет $S = 20$ камней. Тогда после первого хода Пети в куче будет 21 камень или 100 камней. В обоих случаях Ваня увеличивает количество камней в 5 раз и выигрывает в один ход.

2. Возможные значения S : 4, 19. В этих случаях Петя, очевидно, не может выиграть первым ходом. Однако он может получить кучу из 20 камней (при $S = 4$ он увеличивает количество камней в 5 раз; при $S = 19$ – добавляет 1 камень). Эта позиция разобрана в п. 1 б). В ней игрок, который будет ходить (теперь это Ваня), выиграть не может, а его противник (то есть Петя) следующим ходом выигрывает.

3. Возможное значение S : 18. После первого хода Пети в куче будет 19 или 90 камней. Если в куче станет 90 камней, Ваня увеличит количество камней в 5 раз и выигрывает своим первым ходом. Ситуация, когда в куче 19 камней, разобрана в п. 2. В этой ситуации игрок, который будет ходить (теперь это Ваня), выигрывает своим вторым ходом.

В таблице изображено дерево возможных партий при описанной стратегии Вани. Заключительные позиции (в них выигрывает Ваня) подчеркнуты. На рисунке это же дерево изображено в графическом виде (оба способа изображения дерева допустимы).

Исходная позиция	Положения после очередных ходов			
	1-й ход Пети (разобраны все ходы)	1-й ход Вани (только ход по стратегии)	2-й ход Пети (разобраны все ходы)	2-й ход Вани (только ход по стратегии)
18	$18+1=19$	$19+1=20$	$20+1=21$	$21*5=105$
	$18*5=90$	$90*5=450$	$20*5=100$	$100+1=101$

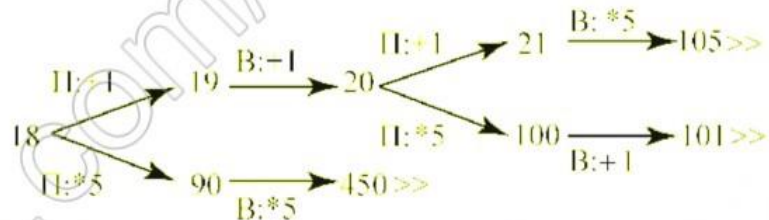


Рис.1. Дерево всех партий, возможных при Ваниной стратегии. Знаком >> обозначены позиции, в которых партия заканчивается

В задаче от ученика требуется выполнить 3 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.

Первое задание считается выполненным полностью, если выполнены полностью оба пункта а) и б). Пункт а) считается выполненным полностью, если правильно указаны все позиции, в которых Петя выигрывает первым ходом, и указано, каким должен быть первый ход. Пункт б) считается выполненным, если правильно указана позиция, в которой Ваня выигрывает первым ходом, и описана стратегия Вани, т.е. показано, как Ваня может получить кучу, в которой содержится нужное количество камней, при любом ходе Пети.

Первое задание считается выполненным частично, если а) правильно указаны все позиции, в которых Петя выигрывает первым ходом, б) правильно указана позиция, в которой Ваня выигрывает первым ходом, и явно сказано, что при любом ходе Пети Ваня может получить кучу, которая содержит нужное для выигрыша количество камней. Отличие от полного решения в том, что явно не указаны ходы, ведущие к выигрышу.

Второе задание выполнено, если правильно указаны обе позиции, выигрышные для Пети, и описана соответствующая стратегия Пети – так, как это написано в примере решения, или другим способом, например с помощью дерева всех партий, возможных при выбранной стратегии Пети.

Третье задание выполнено, если правильно указана позиция, выигрышная для Вани, и построено дерево всех партий, возможных при Ваниной стратегии. Должно быть явно сказано, что в этом дереве в каждой позиции, где должен ходить Петя, разобраны все возможные ходы, а для позиций, где должен ходить Ваня, – только ход, соответствующий стратегии, которую выбрал Ваня.

Во всех случаях стратегии могут быть описаны так, как это сделано в примере решения, или другим способом.

Указания по оцениванию	Баллы
Выполнены второе и третье задания. Первое задание выполнено полностью или частично. Здесь и далее в решениях допускаются арифметические ошибки, которые не искажают сути решения и не приводят к неправильному ответу.	3
Не выполнены условия, позволяющие поставить 3 балла, и выполнено одно из следующих условий. 1. Задание 3 выполнено полностью. 2. Первое и второе задания выполнены полностью. 3. Первое задание выполнено полностью или частично; для заданий 2 и 3 указаны правильные значения S .	2
Не выполнены условия, позволяющие поставить 3 или 2 балла, и выполнено одно из следующих условий. 1. Первое задание выполнено полностью. 2. Во втором задании правильно указано одно из двух возможных значений S , и для этого значения указана и обоснована выигрышная стратегия Пети. 3. Первое задание выполнено частично, и для одного из остальных заданий правильно указано значение S . 4. Для второго и третьего заданий правильно указаны значения S .	1
Не выполнено ни одно из условий, позволяющих поставить 3, 2 или 1 балл.	0
Максимальный балл	3

C4 Соревнования по игре «Тетрис-онлайн» проводятся по следующим правилам.

1. Каждый участник регистрируется на сайте игры под определённым игровым именем. Имена участников не повторяются.
 2. Чемпионат проводится в течение определённого времени. В любой момент этого времени любой зарегистрированный участник может зайти на сайт чемпионата и начать зачётную игру. По окончании игры её результат (количество набранных очков) фиксируется и заносится в протокол.
 3. Участники имеют право играть несколько раз. Количество попыток одного участника не ограничивается.
 4. Окончательный результат участника определяется по одной игре, лучшей для данного участника.
 5. Более высокое место в соревнованиях занимает участник, показавший лучший результат.
 6. При равенстве результатов более высокое место занимает участник, раньше показавший лучший результат.
- В ходе соревнований заполняется протокол, каждая строка которого описывает одну игру и содержит результат участника и его игровое имя. Протокол формируется в реальном времени по ходу проведения чемпионата, поэтому строки в нём расположены в порядке проведения игр: чем раньше встречается строка в протоколе, тем раньше закончилась соответствующая этой строке игра.

Напишите эффективную, в том числе по памяти, программу, которая по данным протокола определяет победителя и призёров. Гарантируется, что в чемпионате участвует не менее трёх игроков.

Перед текстом программы кратко опишите алгоритм решения задачи и укажите используемый язык программирования и его версию.

Описание входных данных

Первая строка содержит число N – общее количество строк протокола.

Каждая из следующих N строк содержит записанные через пробел результат участника (целое неотрицательное число, не превышающее 100 миллионов) и игровое имя (имя не может содержать пробелов). Строки исходных данных соответствуют строкам протокола и расположены в том же порядке, что и в протоколе.

Гарантируется, что количество участников соревнований не меньше 3.

Описание выходных данных

Программа должна вывести имена и результаты трёх лучших игроков по форме, приведённой ниже в примере.

Пример входных данных:

```
9
69485 Jack
95715 qwerty
95715 Alex
83647 M
197128 qwerty
95715 Jack
93289 Alex
95715 Alex
95715 M
```

Пример выходных данных для приведённого выше примера входных данных:

```
1 место. qwerty (197128)
2 место. Alex (95715)
3 место. Jack (95715)
```

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

Программа читает входные данные, не запоминая в массиве информацию обо всех сделанных попытках. Призёров можно определить непосредственно в процессе ввода, запоминая и обновляя 3 лучших результата, но, в отличие от стандартной задачи поиска в последовательности трёх наибольших значений, здесь нужно фиксировать и правильно обрабатывать ситуацию, когда один из текущих призёров в очередной попытке улучшает свой результат. Пример такого решения приведён ниже на алгоритмическом языке.

Вместо отдельных переменных для хранения имён и результатов призёров допускается использование массивов из 3 элементов.

Пример правильной и эффективной программы на алгоритмическом языке

алг

нач

лит имя1="", имя2="", имя3="" | имена победителей
цел сум1=0, сум2=0, сум3=0 | результаты победителей

цел Nлит имяцел сумввод Nнц N разввод сум, имявыборпри имя=имя1:если сум>сум1 то сум1:=сум всепри имя=имя2:выборпри сум>сум1:

имя2:=имя1; сум2:=сум1

имя1:=имя; сум1:=сум

при сум>сум2:

сум2:=сум

всепри сум>сум1:

имя3:=имя2; сум3:=сум2

имя2:=имя1; сум2:=сум1

имя1:=имя; сум1:=сум

при сум>сум2:

имя3:=имя2; сум3:=сум2

имя2:=имя; сум2:=сум

при сум>сум3:

имя3:=имя; сум3:=сум

всекцвывод нс, "1 место. ", имя1, " (", сум1, ")"вывод нс, "2 место. ", имя2, " (", сум2, ")"вывод нс, "3 место. ", имя3, " (", сум3, ")"кон

Указания по оцениванию	Баллы
Программа правильно работает для любых входных данных произвольного размера и находит ответ, не сохраняя информацию обо всех попытках. Допускается наличие в тексте программы одной синтаксической ошибки: пропущен или неверно указан знак пунктуации, неверно написано или пропущено зарезервированное слово языка программирования, не описана или неверно описана переменная, применяется операция, недопустимая для соответствующего типа данных (если одна и та же ошибка встречается несколько раз, то это считается за одну ошибку).	4
Программа работает верно, но размер используемой памяти зависит от количества исходных данных. Например, входные данные запоминаются в массиве или другой структуре данных, размер которой соответствует общему количеству попыток. Допускается одна из следующих ошибок (если одна и та же ошибка повторяется несколько раз, она считается за одну). 1. Неверный ввод исходных данных. 2. Неверно или неполно оформляется вывод результатов. 3. Неверно определяется порядок мест при равных результатах. Допускается наличие от одной до трёх синтаксических ошибок, описанных в критериях на 4 балла.	3
Программа работает в целом верно, эффективно или нет. В реализации алгоритма допущено более 1 ошибки из числа перечисленных в предыдущем пункте или допущены другие ошибки, приводящие к неверной работе программы в отдельных случаях. 2 балла также ставятся за программу, которая находит 3 лучших результата, не учитывая, что некоторые из них могут принадлежать одному игроку, то есть в некоторых ситуациях может присвоить одному игроку сразу несколько мест. Допускается наличие от одной до пяти синтаксических ошибок, описанных в критериях на 4 балла.	2
Программа работает в отдельных частных случаях. Один балл также ставится, если программа написана неверно, но из описания алгоритма и общей структуры программы видно, что экзаменуемый в целом правильно представляет путь решения задачи.	1
Максимальный балл	
	4

Ответы к заданиям с выбором ответа

№ задания	Ответ
A1	1
A2	2
A3	2
A4	3
A5	1
A6	4
A7	4

№ задания	Ответ
A8	4
A9	2
A10	4
A11	1
A12	1
A13	2

Ответы к заданиям с кратким ответом

№ задания	Ответ
B1	21121
B2	70
B3	4
B4	48
B5	55
B6	48
B7	1
B8	104

№ задания	Ответ
B9	18
B10	A36
B11	DFDH
B12	184
B13	88
B14	20
B15	31

Ответы к заданиям с выбором ответа

№ задания	Ответ
A1	2
A2	2
A3	4
A4	3
A5	2
A6	4
A7	4

№ задания	Ответ
A8	2
A9	3
A10	2
A11	1
A12	1
A13	3

Ответы к заданиям с кратким ответом

№ задания	Ответ
B1	12121
B2	75
B3	5
B4	56
B5	110
B6	105
B7	1
B8	108

№ задания	Ответ
B9	18
B10	A12
B11	DEDH
B12	255
B13	88
B14	25
B15	31