

ЕГЭ по информатике 08.05.2014. Досрочная волна, резервный день. Вариант 202.

При выполнении заданий с кратким ответом впишите в поле для ответа цифру, которая соответствует номеру правильного ответа, или число, слово, последовательность букв (слов) или цифр. Ответ следует записывать без пробелов и каких-либо дополнительных символов. Дробную часть отделяйте от целой десятичной запятой. Единицы измерений писать не нужно.

Если вариант задан учителем, вы можете вписать или загрузить в систему ответы к заданиям с развернутым ответом. Учитель увидит результаты выполнения заданий с кратким ответом и сможет оценить загруженные ответы к заданиям с развернутым ответом. Выставленные учителем баллы отобразятся в вашей статистике.

Тестирование в эмуляторе

1. Требовалось написать программу, при выполнении которой с клавиатуры считывается координата точки на прямой (x — действительное число) и определяется принадлежность этой точки одному из выделенных отрезков В и D (включая границы). Программист торопился и написал программу неправильно. (Ниже для Вашего удобства программа представлена на четырёх языках программирования.)



Бейсик	Паскаль
<pre> INPUT x IF x<13 THEN IF x<=7 THEN IF x>=1 THEN PRINT "принадлежит" ELSE PRINT "не принадле- жит" ENDIF ENDIF ENDIF END </pre>	<pre> var x: real; begin readln(x); if x<13 then if x<=7 then if x>=1 then write('принадлежит') else write('не принадле- жит') end. </pre>
Си++	Алгоритмический
<pre> int main(void) { float x; cin >> x; if(x<13) if(x<=7) if(x>=1) cout << "принадле- жит"; else cout << "не принадле- жит"; } </pre>	<pre> алг нач вещ x ввод x если x<13 то если x<=7 то если x>=1 то вывод 'принадлежит' иначе вывод 'не принадле- жит' все все все кон </pre>

Python

```
x = int(input())
if x < 13:
    if x <= 7:
        if x >= 1:
            print("принадлежит")
        else:
            print("не принадлежит")
```

Последовательно выполните следующее.

1. Перерисуйте и заполните таблицу, которая показывает, как работает программа при аргументах, принадлежащих различным областям (А, В, С, D и E). Границы (точки 1, 7, 13 и 19) принадлежат заштрихованным областям (В и D соответственно).

Область	Условие 1 ($x < 13$)	Условие 2 ($x \leq 7$)	Условие 3 ($x \geq 1$)	Программа выведет	Область обрабатывается верно
А					
В					
С					
D					
E					

В столбцах условий укажите «да», если условие выполнится; «нет», если условие не выполнится; «—» (прочерк), если условие не будет проверяться; «не изв.», если программа ведёт себя по-разному для разных значений, принадлежащих данной области. В столбце «Программа выведет» укажите, что программа выведет на экран. Если программа ничего не выводит, поставьте «—» (прочерк). Если для разных значений, принадлежащих области, будут выведены разные тексты, напишите «не изв.». В последнем столбце укажите «Да» или «Нет».

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.)

2. Дан целочисленный массив из 30 элементов. Элементы массива могут принимать значения от 0 до 100 и содержат итоговые баллы участников олимпиады по информатике. Опишите на русском языке или на одном из языков программирования алгоритм, который позволяет подсчитать и вывести средний балл среди участников, ставших призёрами олимпиады (призёрами олимпиады стали школьники, набравшие более 50 баллов). Гарантируется, что хотя бы один участник олимпиады стал её призёром. Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать часть из них.

Бейсик	Паскаль
<pre> N=30 DIM A(N) AS INTEGER DIM I, X, Y AS INTEGER DIM S AS SINGLE FOR I = 1 TO N INPUT A(I) NEXT I ... END </pre>	<pre> const N=30; var a: array [1..N] of integer; i, x, y: integer; s: real; begin for i:=1 to N do read (a[i]); ... end. </pre>
Си++	Алгоритмический язык
<pre> #include <iostream> using namespace std; #define N 30 int main(void) {int a[N]; int i, x, y; float s; for (i=0; i < N; i++) cin >> a[i]; ... } </pre>	<pre> алг нач цел N = 30 целтаб a[1:N] цел i, x, y вещ s нц для i от 1 до N ввод a[i] кц ... кон </pre>
Естественный язык	
<p>Объявляем массив А из 30 элементов. Объявляем целочисленные переменные I, X, Y. Объявляем вещественную переменную S. В цикле от 1 до 30 вводим элементы массива А с 1-го по 30-й.</p>	
Python	
<pre> # допускается также # использовать две # целочисленные переменные x, y # и вещественную переменную s a = [] n = 30 for i in range(0, n): a.append(int(input())) ... </pre>	

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 Borland Pascal 7.0) или в виде блок-схемы. В этом случае Вы должны использовать переменные, аналогичные переменным, используемым в алгоритме, записанном на естественном языке, с учётом синтаксиса и особенностей используемого Вами языка программирования.

3. 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 два раза. Например, имея кучу из 15 камней, за один ход можно получить кучу из 16 или 30 камней. Для того чтобы делать ходы, у каждого игрока есть неограниченное количество камней.

Игра завершается в тот момент, когда количество камней в куче становится не менее 94. Победителем считается игрок, сделавший последний ход, т. е. первым получивший кучу, в которой будет 94 или больше камней.

В начальный момент в куче было S камней; $1 \leq S \leq 93$.

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.

Выполните следующие задания. Во всех случаях обосновывайте свой ответ.

Задание 1

а) Укажите все такие значения числа S , при которых Петя может выиграть в один ход. Обоснуйте, что найдены все нужные значения S , и укажите выигрышающие ходы.

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.

Задание 2

Укажите два таких значения S , при которых у Пети есть выигрышная стратегия, причём одновременно выполняются два условия:

- Петя не может выиграть за один ход;
- Петя может выиграть своим вторым ходом независимо от того, как будет ходить Ваня.

Для каждого указанного значения S опишите выигрышную стратегию Пети.

Задание 3

Укажите значение S , при котором одновременно выполняются два условия:

- у Вани есть выигрышная стратегия, позволяющая ему выиграть первым или вторым ходом при любой игре Пети;
- 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 (в виде рисунка или таблицы). На рисунке на рёбрах дерева указывайте, кто делает ход; в узлах — количество камней в позиции.

4. На вход программе подаются сведения о пассажирах, забронировавших по Интернету авиабилеты (только тех, у кого время бронирования ещё не истекло). В первой строке задано текущее время: через двоеточие два целых числа, соответствующие часам (от 00 до 23 — ровно 2 символа) и минутам (от 00 до 59 — ровно 2 символа). Во второй строке сообщается число пассажиров N , которое не меньше 3, но не превосходит 1000. Каждая из следующих N строк имеет следующий формат:

<Фамилия> <время окончания брони>,

где <Фамилия> — строка, состоящая не более чем из 20 непробельных символов; <время окончания брони> — через двоеточие два целых числа, соответствующие часам (от 00 до 23 — ровно 2 символа) и минутам (от 00 до 59 — ровно 2 символа). <Фамилия> и <время окончания брони> разделены одним пробелом. Сведения отсортированы в порядке времени, когда производилось бронирование. Все значения времени относятся к текущим суткам.

Требуется написать эффективную программу (укажите используемую версию языка программирования, например Borland Pascal 7.0), которая в хронологическом порядке (т. е. в порядке возрастания значения времени окончания брони) выведет фамилии пассажиров, у которых в ближайшие 3 часа закончится бронь.

Пример входных данных:

10:00

3

Иванов 13:00

Петров 10:00

Сидоров 13:12

Результат работы программы на этих входных данных:

Петров

Иванов