**Промежуточная аттестация по информатике и ИКТ   
10 класс**

**Демо-вариант**

|  |
| --- |
| **Часть А. При вы­пол­не­нии заданий 1—13 ответом яв­ля­ет­ся одна цифра, ко­то­рая соответствует но­ме­ру правильного ответа или число, по­сле­до­ва­тель­ность букв или цифр. Ответ сле­ду­ет записывать без про­бе­лов и каких-либо до­пол­ни­тель­ных символов в бланк ответов №1.** |

**А1.** Укажите целое число от 7 до 10, двоичная запись которого содержит ровно два значащих нуля. Если таких чисел несколько, укажите наибольшее из них.

**А2.** Логическая функция *F* задаётся выражением *(x ∨ y) → (z ≡ x)*.

Дан частично заполненный фрагмент, содержащий **неповторяющиеся** строки таблицы истинности функции *F*.

Определите, какому столбцу таблицы истинности соответствует каждая из переменных *x*, *y*, *z*.

|  |  |  |  |
| --- | --- | --- | --- |
| **Переменная 1** | **Переменная 2** | **Переменная 3** | **Функция** |
| ??? | ??? | ??? | *F* |
|  | 0 | 0 | 0 |
|  | 0 |  | 0 |

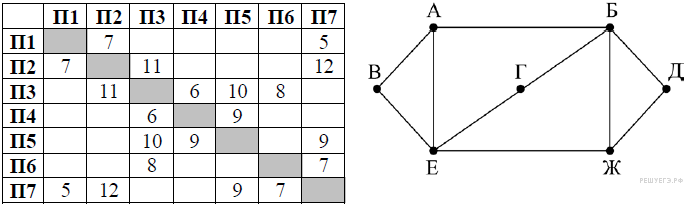
В ответе напишите буквы *x*, *y*, *z* в том порядке, в котором идут соответствующие им столбцы (сначала – буква, соответствующая первому столбцу; затем – буква, соответствующая второму столбцу, и т. д.). Буквы в ответе пишите подряд, никаких разделителей между буквами ставить не нужно.

*Пример.* Пусть задано выражение*x → y,* зависящее от двух переменных *x* и *y*, и фрагмент таблицы истинности:

|  |  |  |
| --- | --- | --- |
| **Переменная 1** | **Переменная 2** | **Функция** |
| ??? | ??? | *F* |
| 0 | 1 | 0 |

Тогда первому столбцу соответствует переменная *y*, а второму столбцу соответствует переменная *x*. В ответе нужно написать: *yx*.

**А3.** На рисунке схема дорог изображена в виде графа, в таблице содержатся сведения о длине этих дорог в километрах.



Так как таблицу и схему рисовали независимо друг от друга, нумерация населённых пунктов в таблице никак не связана с буквенными обозначениями на графе. Определите длину кратчайшего пути из пункта В в пункт Д, если передвигаться можно только по указанным дорогам. В ответе запишите целое число — длину пути в километрах.

**А4.** 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. На основании приведённых данных определите, сколько всего внуков и внучек у Кравец Д.К.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  | | --- | | **Таблица 1** | | ID | Фамилия\_И.О. | Пол | | 866 | Кравец Д.К. | Ж | | 867 | Тошич Б.Ф. | М | | 879 | Гонтарь В.А. | Ж | | 885 | Крон К.Г. | М | | 900 | Кислюк Л.А. | Ж | | 904 | Петраш А.И. | М | | 911 | Тошич А.Б. | Ж | | 932 | Петраш П.А. | Ж | | 938 | Тошич И.Б. | М | | 949 | Седых Г.Р. | Ж | | 970 | Кислюк А.П. | М | | 995 | Тошич Т.И. | Ж | | 1017 | Тошич П.И. | М | | 1026 | Мухина Р.Г. | Ж | | 1041 | Сайко М.А. | Ж | | 1056 | Кислюк П.А. | М | | ... | ... | ... | | |  | | --- | | **Таблица 2** | | ID\_Родителя | ID\_Ребенка | | 866 | 911 | | 866 | 938 | | 867 | 911 | | 867 | 938 | | 911 | 879 | | 911 | 1041 | | 904 | 900 | | 938 | 995 | | 938 | 1017 | | 949 | 995 | | 949 | 1017 | | 970 | 879 | | 970 | 1041 | | 904 | 932 | | 1026 | 900 | | 1026 | 932 | | ... | ... | |

**А5.** Для кодирования некоторой последовательности, состоящей из букв А, Б, В, Г и Д, используется неравномерный двоичный код, позволяющий однозначно декодировать полученную двоичную последовательность. Вот этот код: А — 1; Б — 0100; В — 000;   
Г — 011; Д — 0101. Требуется сократить для одной из букв длину кодового слова так, чтобы код по-прежнему можно было декодировать однозначно. Коды остальных букв меняться не должны. Каким из указанных способов это можно сделать?

1) для буквы Г — 11

2) для буквы В — 00

3) для буквы Г — 01

4) это невозможно

**А6.**  Автомат обрабатывает натуральное число *N* по следующему алгоритму:

1. Строится двоичная запись числа *N*.

2. Складываются все цифры полученной двоичной записи. В конец записи (справа) дописывается остаток от деления суммы на 2.

3. Предыдущий пункт повторяется для записи с добавленной цифрой.

4. Результат переводится в десятичную систему и выводится на экран.

Пример. Дано число *N* = 13. Алгоритм работает следующим образом:

1. Двоичная запись числа *N*: 1101.

2. Сумма цифр двоичной записи 3, остаток от деления на 2 равен 1, новая запись 11011.

3. Сумма цифр полученной записи 4, остаток от деления на 2 равен 0, новая запись 110110.

4. На экран выводится число 54.

Какое наименьшее число, большее 93, может появиться на экране в результате работы автомата?

**А7.** Определите число, которое будет напечатано в результате выполнения программы (записанной ниже на разных языках программирования).

|  |  |
| --- | --- |
| **Бейсик** | **Паскаль** |
| **DIM** N, S **AS** **INTEGER**  N = 20  S = 0  **WHILE** S <= 257  S = S + 10  N = N + 3  **WEND**  **PRINT** N | **var** n, s: **integer**;  **begin**      n := 20;      s := 0;  **while** s <= 257 **do**  **begin**          s := s + 10;          n := n + 3  **end**;  **write**(n)  **end**. |
| **Си++** | **Алгоритмический язык** |
| #include <iostream>  using namespace std;  **int** main()  **{**  **int** n, s;      n = 20;      s = 0;  **while** (s <= 257)  **{**          s = s + 10;          n = n + 3;  **}**      cout « n « endl;  **}** | **алг**  **нач**  **цел** n, s  n := 20  s := 0  **нц** **пока** s <= 257      s := s + 10      n := n + 3  **кц**  **вывод** n  **кон** |
| **Python** | |
| n = 20  s = 0  while s <= 257:      s += 10      n += 3  print(n) | |

**А8.** Документ объёмом 16 Мбайт можно передать с одного компьютера на другой двумя способами:

А. Сжать архиватором, передать архив по каналу связи, распаковать.

Б. Передать по каналу связи без использования архиватора.

Какой способ быстрее и на сколько, если

· средняя скорость передачи данных по каналу связи составляет 220 бит в секунду;

· объём сжатого архиватором документа равен 25% от исходного;

· время, требуемое на сжатие документа, — 18 секунд, на распаковку — 2 секунды?

В ответе напишите букву А, если способ А быстрее, или Б, если быстрее способ Б. Сразу после буквы напишите, на сколько секунд один способ быстрее другого. Например, если способ Б быстрее способа А на 23 секунды, в ответе нужно написать Б23. Слов «секунд», «сек.», «с» к ответу добавлять не нужно.

**А9.** 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. Адрес сети получается в результате применения поразрядной конъюнкции к заданному IP-адресу узла и маске. По заданным IP-адресу сети и маске определите адрес сети.

IP-адрес узла: 224.24.254.134

Маска: 255.255.240.0

При записи ответа выберите из приведённых в таблице чисел четыре элемента IP-адреса и запишите в нужном порядке соответствующие им буквы без использования точек.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| A | B | C | D | E | F | G | H |
| 255 | 240 | 232 | 224 | 234 | 24 | 8 | 0 |

*Пример. Пусть искомый IP-адрес: 192.168.128.0, и дана таблица:*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| A | B | C | D | E | F | G | H |
| 128 | 168 | 255 | 8 | 127 | 0 | 17 | 192 |

*В этом случае правильный ответ будет записан в виде: HBAF.*

**А10.** Исполнитель Редактор получает на вход строку цифр и преобразует 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 85 идущих подряд цифр 7? В ответе запишите полученную строку.

НАЧАЛО

ПОКА **нашлось** (333) ИЛИ **нашлось** (777)

ЕСЛИ **нашлось** (333)

ТО **заменить** (333, 7)

ИНАЧЕ **заменить** (777, 3)

КОНЕЦ ЕСЛИ

КОНЕЦ ПОКА

КОНЕЦ

**А11.** Запись числа 2310 в некоторой системе счисления выглядит так: **212q.** Найдите основание системы счисления q.

**А12.** В языке запросов поискового сервера для обозначения логической операции «ИЛИ» используется символ «|», а для логической операции «И» — символ «&». В таблице приведены запросы и количество найденных по ним страниц некоторого сегмента сети Интернет:

|  |  |
| --- | --- |
| **Запрос** | **Найдено страниц**  **(в тысячах)** |
| Китай & (Япония | США) | 390 |
| Китай & Япония | 180 |
| Китай & США & Япония | 50 |

Какое количество страниц (в тыс.) будет найдено по запросу

*Китай & США?*

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

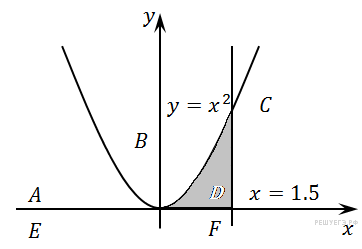
**А 13.** Ниже на четырёх языках записан алгоритм. Получив на вход число x, этот алгоритм печатает два числа: a и b. Укажите наименьшее из таких чисел x, при вводе которого алгоритм печатает сначала 3, а потом 2.

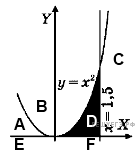
|  |  |
| --- | --- |
| **Бейсик** | **Паскаль** |
| **DIM** X, A, B **AS** **INTEGER**  **INPUT** X  A=0: B=0  **WHILE** X > 0      A = A+1  **IF** B < (X MOD 8) **THEN**          B = X MOD 8  **END** **IF**      X = X \ 8  **WEND**  **PRINT** A  **PRINT** B | **var** x, a, b: **integer**;  **begin**  **readln**(x);      a:=0; b:=0;  **while** x>0 **do**  **begin**              a:=a + 1;  **if** b < (x **mod** 8)  **then**                  b:=x **mod** 8;              x:=x **div** 8;  **end**;  **writeln**(a); **write**(b);  **end**. |
| **Си++** | **Алгоритмический** |
| #include <iostream>  using namespace std;  **int** main()  **{**  **int** x, a, b;      cin >> x;      a=0; b=0;  **while** (x>0)**{**          a = a+1;  **if** (b < (x%8)**{**              b = x%8;  **}**          x = x/8;  **}**      cout << a << endl << b << endl;  **}** | **алг**  **нач**  **цел** x, a, b  **ввод** x  a:=0; b:=0  **нц** **пока** x>0      a:=a+1  **если** b < mod(x,8)  **то**              b:=mod(x,8)  **все**      x:=div(x,8)  **кц**  **вывод** a, нс, b  **кон** |
| **Python** | |
| x = **int**(input())  a = 0  b = 0  **while** x > 0:      a += 1  **if** b < (x % 8):          b =x % 8      x *//= 8*  print(a)  print(b) | |

|  |
| --- |
| **Часть В.** Для записи ответов на задания этой части используйте бланк ответов № 2. Запишите сначала номер задания (В1, В2 и т. д.), а затем полное решение. Ответы записывайте чётко и разборчиво. |

**В1.** Требовалось написать программу, которая вводит с клавиатуры координаты точки на плоскости (х, у — действительные числа) и определяет принадлежность точки заштрихованной области, включая ее границы.

|  |  |
| --- | --- |
| **Паскаль** | **Бейсик** |
| **var** х,у: **real**;  **begin**  **readln**(х,у);  **if** у<=х\*х **then**  **if** х<=1.5 **then**  **if** y>=0 **then**  **write**('принадлежит ')  **else**  **write** ('не принадлежит');  **end**. | **INPUT** x, у  **IF** y<=x\*x **THEN**  **IF** x<=1.5 **THEN**  **IF** y>=0 **THEN**  **PRINT** "принадлежит"  **ELSE**  **PRINT** "не принадлежит"  **ENDIF**  **ENDIF**  **ENDIF**  **END** |
| **Си++** | **Алгоритмический** |
| **int** main(**void**)  **{** **float** x,у;  cin >> x >> y;  **if** (y<=x\*x)  **if** (x<=l.5)  **if** (y>=0)  cout << "принадлежит";  **else**  cout << "не принадлежит";  **}** | **алг**  **нач**  **вещ** x,y  **ввод** x,y  **если** у<=х\*х **то**  **если** х<=1.5 **то**  **если** y>=0 **то**  **вывод** 'принадлежит'  **иначе**  **вывод** 'не принадлежит'  **все**  **все**  **все**  **кон** |
| **Python** | |
| x = float(input())  y = float(input())  if y<=x\*x:      if x<=1.5:          if y>=0:              print("принадлежит")          else:              print("не принадлежит") | |



Последовательно выполните следующее:

1. Перерисуйте и заполните таблицу, которая показывает, как работает программа при аргументах, принадлежащих различным областям (A, B, C, D, E и F).

Точки, лежащие на границах областей, отдельно не рассматривать. В столбцах условий укажите "да", если условие выполнится, "нет", если условие не выполнится, "—" (прочерк), если условие не будет проверяться, "не изв.", если программа ведет себя по-разному для разных значений, принадлежащих данной области. В столбце "Программа выведет" укажите, что программа выведет на экран. Если программа ничего не выводит, напишите "—" (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 ее неправильной работы. (Это можно сделать несколькими способами, достаточно указать любой способ доработки исходной программы.)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Область | Условие 1  (у<=x\*x) | Условие 2  (x<=1.5) | Условие 3  (y>=0) | Программа выведет | Область обрабатывается  верно |
| A |  |  |  |  |  |
| В |  |  |  |  |  |
| С |  |  |  |  |  |
| D |  |  |  |  |  |
| Е |  |  |  |  |  |
| F |  |  |  |  |  |

**В2.** 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 **три 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), (30, 7), (10, 8), (10, 21). 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 68. Победителем считается игрок, сделавший последний ход, т.е. первым получивший такую позицию, при которой в кучах будет 68 или больше камней.

В начальный момент в первой куче было шесть камней, во второй куче — *S* камней; 1 ≤ *S* ≤ 61.

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. В описание выигрышной стратегии не следует включать ходы играющего по этой стратегии игрока, не являющиеся для него безусловно выигрышными, т.е. не являющиеся выигрышными независимо от игры противника.

Выполните следующие задания.

**Задание 1.**

а) Укажите все такие значения числа *S*, при которых Петя может выиграть за один ход.

б) Известно, что Ваня выиграл своим первым ходом после неудачного первого хода Пети. Укажите минимальное значение *S*, когда такая ситуация возможна.

**Задание 2.** Укажите такое значение *S*, при котором 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 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).

В узлах дерева указывайте позиции, на рёбрах рекомендуется указывать ходы. Дерево не должно содержать партии, невозможные при реализации выигрывающим игроком своей выигрышной стратегии. Например, полное дерево игры не является верным ответом на это задание.