

**Тематическая диагностическая работа
по подготовке к ОГЭ**

**по ИНФОРМАТИКЕ
по теме**

«Логика и алгоритмы»

**1 декабря 2014 года
9 класс**

Вариант ИН90301

Район	
Город (населённый пункт)	
Школа	
Класс	
Фамилия	
Имя	
Отчество	

Инструкция по выполнению работы

На выполнение диагностической работы по информатике отводится 90 минут. Работа включает в себя 9 заданий. Задание 9.1 или 9.2 выполняется по Вашему выбору.

Ответы к заданиям 1 и 3 записываются в виде одной цифры, которая соответствует номеру правильного ответа. Эту цифру запишите в поле ответа в тексте работы.

Ответы к заданиям 2, 4–8 записываются в виде последовательности букв или числа в поле ответа в тексте работы. В случае записи неверного ответа зачеркните его и запишите рядом новый.

Задания 9.1 или 9.2 выполняются на компьютере. Результатом выполнения задания 9.1 или 9.2 является файл, который необходимо сохранить под именем и в формате, указанном учителем.

Советуем выполнять задания в том порядке, в котором они даны. В целях экономии времени пропускайте задание, которое не удаётся выполнить сразу, и переходите к следующему. Если после выполнения всей работы у Вас останется время, то Вы сможете вернуться к пропущенным заданиям. Постарайтесь выполнить как можно больше заданий.

Желаем успеха!

Часть 1

- 1 Для какого из приведённых отрезков числовой прямой **ложно** высказывание: (все принадлежащие отрезку числа являются отрицательными) ИЛИ НЕ (длина отрезка больше 5)?

1) [6; 10] 2) [-2; 4] 3) [-15; -5] 4) [-7; -3]

Ответ:

- 2 Сообщения, в которые могут входить только буквы А, В, Е, К, Н, закодированы азбукой Морзе без разделителей между буквами. Вот нужный фрагмент азбуки Морзе:

А	В	Е	К	Н
· -	· - -	·	· - -	· ·

Поскольку разделители между буквами отсутствуют, некоторые сообщения можно декодировать разными способами. Например, · - · - может означать ЕК, а может и АА. Вот три сообщения:

· - - - -
- - - - -
- - - - -

Только одно из них декодируется единственным способом. Найдите его и расшифруйте. То, что получилось, запишите в качестве ответа.

Ответ: _____.

- 3 Исполнитель Чертёжник перемещается на координатной плоскости, оставляя след в виде линии. Чертёжник может выполнять команду **Сместиться на (a, b)** (где a, b – целые числа), перемещающую его из точки с координатами (x, y) в точку с координатами (x + a, y + b). Если числа a, b положительные, значение соответствующей координаты увеличивается, если отрицательные – уменьшается. Например, если Чертёжник находится в точке с координатами (3, 1), то команда **Сместиться на (-6, 1)** переместит его в точку (-3, 2).

Запись

Повтори k раз

Команда1 Команда2 Команда3

конец

означает, что последовательность команд **Команда1 Команда2 Команда3** повторится k раз. Чертёжнику был дан для исполнения следующий алгоритм:

Повтори 4 раз

Сместиться на (-1, 0) Сместиться на (2, 1) Сместиться на (2, -2)

конец

Какую команду надо выполнить Чертёжнику, чтобы вернуться в исходную точку, из которой он начал движение?

- 1) Сместиться на (-12, 4) 3) Сместиться на (3, -1)
2) Сместиться на (12, -4) 4) Сместиться на (-3, 1)

Ответ:

- 4 В программе «:=» обозначает оператор присваивания, знаки «+», «-», «*» и «/» – соответственно операции сложения, вычитания, умножения и деления. Правила выполнения операций и порядок действий соответствуют правилам арифметики.

Определите значение переменной a после выполнения алгоритма:

```
a := 3
b := 6
b := a + b - b/a
a := a * b - a - b
```

В ответе укажите одно целое число – значение переменной a.

Ответ: _____.

- 5 Запишите значение переменной s, полученное в результате работы следующей программы. Текст программы приведён на трёх языках программирования.

Алгоритмический язык	Бейсик	Паскаль
<u>алг</u> <u>нач</u> цел s, k s := 3 нц для k от 5 до 15 s := s+5 кц вывод s кон	DIM k,s AS INTEGER s = 3 FOR k = 5 TO 15 s = s+5 NEXT k PRINT s	Var s,k: integer; Begin s := 3; for k:=5 to 15 do s := s+5; writeln(s); End.

Ответ: _____.

6

В таблице Dat хранятся данные о количестве пасмурных дней в каждом месяце прошлого года в городе Y (Dat [1] – число пасмурных дней в январе, Dat [2] – в феврале и т. д.). Определите, какое число будет напечатано в результате работы следующей программы. Текст программы приведён на трёх языках программирования.

Паскаль	Бейсик
<pre> Var k, m: integer; Dat: array[1..12] of integer; Begin Dat[1] := 12; Dat[2] := 8; Dat[3] := 14; Dat[4] := 16; Dat[5] := 10; Dat[6] := 21; Dat[7] := 15; Dat[8] := 15; Dat[9] := 12; Dat[10] := 8; Dat[11] := 7; Dat[12] := 5; m := 0; for k := 1 to 12 do if Dat[k] > 10 then begin m := m + 1 end; writeln(m) end. </pre>	<pre> DIM Dat(12) AS INTEGER DIM k,m AS INTEGER Dat(1) = 12 Dat(2) = 8 Dat(3) = 14 Dat(4) = 16 Dat(5) = 10 Dat(6) = 21 Dat(7) = 15 Dat(8) = 15 Dat(9) = 12 Dat(10) = 8 Dat(11) = 7 Dat(12) = 5 m = 0 FOR k = 1 TO 12 IF Dat(k) >10 THEN m = m+1 END IF NEXT k PRINT m </pre>

Алгоритмический язык

```

алг
нач
    целтаб Dat[1:12]
    цел k, m
    Dat[1] := 12
    Dat[2] := 8
    Dat[3] := 14
    Dat[4] := 16
    Dat[5] := 10
    Dat[6] := 21
    Dat[7] := 15
    Dat[8] := 15
    Dat[9] := 12
    Dat[10] := 8
    Dat[11] := 7
    Dat[12] := 5
    m := 0
    нц для k от 1 до 12
        если Dat[k] > 10 то
            m := m+1
        все
    кц
    вывод m
кон

```

Ответ: _____.

7

У исполнителя Уменьшитель две команды, которым присвоены номера:

1) вычти 1

2) раздели на 2

Первая из них уменьшает число на экране на 1, вторая – уменьшает его в 2 раза. Составьте алгоритм получения из числа 100 числа 11, содержащий не более пяти команд. В ответе запишите только номера команд.

Например, 22211 – это алгоритм

раздели на 2

раздели на 2

раздели на 2

вычти 1

вычти 1,

который преобразует число 64 в 6.

Если таких алгоритмов более одного, то запишите любой из них.

Ответ: _____.

8

Некоторый алгоритм из одной цепочки символов получает новую цепочку следующим образом. Сначала вычисляется длина исходной цепочки символов; если она чётна, то в середину цепочки символов добавляется символ **Ч**, а если нечётна, то в начало цепочки добавляется символ **Н**. В полученной цепочке символов каждая буква заменяется буквой, предшествующей ей в русском алфавите (**Б** – на **А**, **В** – на **Б** и т. д., а **А** – на **Я**).

Получившаяся таким образом цепочка является результатом работы алгоритма.

Например, если исходной цепочкой была цепочка **ВРМ**, то результатом работы алгоритма будет цепочка **МБПЛ**, а если исходной цепочкой была **ПД**, то результатом работы алгоритма будет цепочка **ОЦГ**.

Алгоритм применили дважды к некоторой исходной цепочке (то есть его применили к исходной цепочке, а затем вновь применили к результату). В итоге получили цепочку **ЛКЦСТ**. Определите исходную цепочку.

Русский алфавит: **АБВГДЕЁЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ**.

Ответ: _____.

Часть 2

Задание 9.1 или 9.2 выполняется на компьютере. Результатом исполнения задания является отдельный файл (для одного задания – один файл). Формат файла, его имя и каталог для сохранения Вам сообщит учитель.

Выберите для выполнения **ОДНО** из предложенных ниже заданий: 9.1 или 9.2.

9.1

Исполнитель Робот умеет перемещаться по лабиринту, начерченному на плоскости, разбитой на клетки. Ниже приведено описание Робота.

У Робота есть четыре команды перемещения:

вверх

вниз

влево

вправо

При выполнении любой из этих команд Робот перемещается на одну клетку соответственно: вверх ↑, вниз ↓, влево ←, вправо →.

Между соседними (по сторонам) клетками может стоять стена, через которую Робот пройти не может. Если Робот получает команду передвижения через стену, то он разрушается.

Четыре команды проверяют истинность условия отсутствия стены у каждой стороны той клетки, где находится Робот:

сверху свободно

снизу свободно

слева свободно

справа свободно

Эти команды можно использовать вместе с условием **если**, имеющим следующий вид:

если <условие> то

последовательность команд

все

Последовательность команд – это одна или несколько любых команд, выполняемых Роботом. Например, для передвижения на одну клетку вправо, если справа нет стенки, можно использовать такой алгоритм:

если справа свободно то

вправо

все

В одном условии можно использовать несколько команд, применяя логические связки **и**, **или**, **не**, например:

если (справа свободно) и (не снизу свободно) то

вправо

все

Для повторения последовательности команд можно использовать цикл **пока**, имеющий следующий вид:

нц пока <условие>

последовательность команд

кц

Например, для безопасного движения вправо можно использовать следующий алгоритм:

нц пока справа свободно

вправо

кц

Также у Робота есть команда **закрасить**, закрашивающая клетку, в которой Робот находится в настоящий момент.

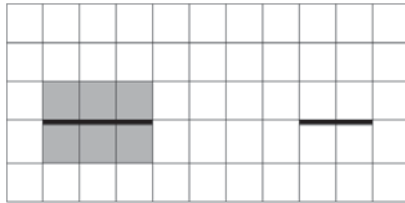
Выполните задание.

На бесконечном поле имеется горизонтальная стена. **Длина стены неизвестна**. В стене есть ровно один проход, точное место прохода и его ширина неизвестны. Робот находится в клетке, расположенной непосредственно над стеной у её правого конца.

На рисунке указан один из возможных способов расположения стены и Робота (Робот обозначен буквой «Р»):



Напишите для Робота алгоритм, закрашивающий все клетки, примыкающие горизонтальными сторонами к участку стены левее прохода. Робот должен закрасить только клетки, удовлетворяющие данному условию. Например, для приведённого выше рисунка Робот должен закрасить следующие клетки (см. рисунок):



Конечное расположение Робота может быть произвольным. Алгоритм должен решать задачу для произвольного размера поля и любого допустимого расположения стен внутри прямоугольного поля. При исполнении алгоритма Робот не должен разрушиться.

Алгоритм напишите в текстовом редакторе и сохраните в текстовом файле.

Название файла и каталог для сохранения Вам сообщит учитель.

9.2 Напишите программу, которая в последовательности натуральных чисел определяет максимальное число, оканчивающееся на 8. Программа получает на вход количество чисел в последовательности, а затем сами числа. В последовательности всегда имеется число, оканчивающееся на 8.

Количество чисел не превышает 1000. Введённые числа не превышают 30 000.

Программа должна вывести одно число – максимальное число, оканчивающееся на 8.

Пример работы программы:

Входные данные	Выходные данные
4 138 81 80 108	138

**Тематическая диагностическая работа
по подготовке к ОГЭ**

**по ИНФОРМАТИКЕ
по теме**

«Логика и алгоритмы»

**1 декабря 2014 года
9 класс**

Вариант ИН90302

Район	
Город (населённый пункт)	
Школа	
Класс	
Фамилия	
Имя	
Отчество	

Инструкция по выполнению работы

На выполнение диагностической работы по информатике отводится 90 минут. Работа включает в себя 9 заданий. Задание 9.1 или 9.2 выполняется по Вашему выбору.

Ответы к заданиям 1 и 3 записываются в виде одной цифры, которая соответствует номеру правильного ответа. Эту цифру запишите в поле ответа в тексте работы.

Ответы к заданиям 2, 4–8 записываются в виде последовательности букв или числа в поле ответа в тексте работы. В случае записи неверного ответа зачеркните его и запишите рядом новый.

Задания 9.1 или 9.2 выполняются на компьютере. Результатом выполнения задания 9.1 или 9.2 является файл, который необходимо сохранить под именем и в формате, указанном учителем.

Советуем выполнять задания в том порядке, в котором они даны. В целях экономии времени пропускайте задание, которое не удаётся выполнить сразу, и переходите к следующему. Если после выполнения всей работы у Вас останется время, то Вы сможете вернуться к пропущенным заданиям. Постарайтесь выполнить как можно больше заданий.

Желаем успеха!

Часть 1

- 1 Для какого из приведённых отрезков числовой прямой **ложно** высказывание: (длина отрезка больше 6) ИЛИ НЕ (все принадлежащие отрезку числа являются положительными)?

1) $[-10; -6]$ 2) $[-3; 1]$ 3) $[7; 10]$ 4) $[3; 11]$

Ответ:

- 2 Сообщения, в которые могут входить только буквы А, В, Е, К, Н, закодированы азбукой Морзе без разделителей между буквами. Вот нужный фрагмент азбуки Морзе:

А	В	Е	К	Н
· -	· - -	·	- · -	· - ·

Поскольку разделители между буквами отсутствуют, некоторые сообщения можно декодировать разными способами. Например, $· - ·$ может означать НЕ, а может и АЕЕ. Вот три сообщения:

· - - - - -
· - - - - -
· - - - - -

Только одно из них декодируется единственным способом. Найдите его и расшифруйте. То, что получилось, запишите в качестве ответа.

Ответ: _____.

- 3 Исполнитель Чертёжник перемещается на координатной плоскости, оставляя след в виде линии. Чертёжник может выполнять команду **Сместиться на (a, b)** (где a, b – целые числа), перемещающую его из точки с координатами (x, y) в точку с координатами (x + a, y + b). Если числа a, b положительные, значение соответствующей координаты увеличивается, если отрицательные – уменьшается. Например, если Чертёжник находится в точке с координатами (3, 1), то команда **Сместиться на (-6, 1)** переместит его в точку (-3, 2).

Запись

Повтори k раз

Команда1 Команда2 Команда3

конец

означает, что последовательность команд **Команда1 Команда2 Команда3** повторится k раз. Чертёжнику был дан для исполнения следующий алгоритм:

Повтори 4 раз

Сместиться на (3, -2) Сместиться на (-1, 1) Сместиться на (-1, 3)

конец

Какую команду надо выполнить Чертёжнику, чтобы вернуться в исходную точку, из которой он начал движение?

- 1) Сместиться на (-1, -2) 3) Сместиться на (4, 8)
2) Сместиться на (1, 2) 4) Сместиться на (-4, -8)

Ответ:

- 4 В программе «:=» обозначает оператор присваивания, знаки «+», «-», «*» и «/» – соответственно операции сложения, вычитания, умножения и деления. Правила выполнения операций и порядок действий соответствуют правилам арифметики.

Определите значение переменной a после выполнения алгоритма:

```
a := 4
b := 8
b := a + b - b/a
a := a * b - a - b
```

В ответе укажите одно целое число – значение переменной a.

Ответ: _____.

- 5 Запишите значение переменной s, полученное в результате работы следующей программы. Текст программы приведён на трёх языках программирования.

Алгоритмический язык	Бейсик	Паскаль
<u>алг</u> <u>нач</u> <u>цел</u> s, k s := 2 <u>нц</u> <u>для</u> k <u>от</u> 7 <u>до</u> 17 s := s+10 <u>кц</u> <u>вывод</u> s <u>кон</u>	DIM k, s AS INTEGER s = 2 FOR k = 7 TO 17 s = s+10 NEXT k PRINT s	Var s, k: integer; Begin s := 2; for k:=7 to 17 do s := s+10; writeln(s); End.

Ответ: _____.

6

В таблице Dat хранятся данные о количестве ясных дней в каждом месяце прошлого года в городе Z (Dat [1] – число ясных дней в январе, Dat [2] – в феврале и т. д.). Определите, какое число будет напечатано в результате работы следующей программы.

Текст программы приведён на трёх языках программирования.

Паскаль	Бейсик
<pre> Var k, m: integer; Dat: array[1..12] of integer; Begin Dat[1] := 11; Dat[2] := 8; Dat[3] := 14; Dat[4] := 16; Dat[5] := 10; Dat[6] := 24; Dat[7] := 15; Dat[8] := 15; Dat[9] := 12; Dat[10] := 8; Dat[11] := 7; Dat[12] := 6; m := 0; for k := 1 to 12 do if Dat[k] < 15 then begin m := m + 1 end; writeln(m) End.</pre>	<pre> DIM Dat(12) AS INTEGER DIM k,m AS INTEGER Dat(1) = 11 Dat(2) = 8 Dat(3) = 14 Dat(4) = 16 Dat(5) = 10 Dat(6) = 24 Dat(7) = 15 Dat(8) = 15 Dat(9) = 12 Dat(10) = 8 Dat(11) = 7 Dat(12) = 6 m = 0 FOR k = 1 TO 12 IF Dat(k) < 15 THEN m = m+1 END IF NEXT k PRINT m</pre>

Алгоритмический язык

```

алг
нач
    целтаб Dat[1:12]
    цел k, m
    Dat[1] := 11
    Dat[2] := 8
    Dat[3] := 14
    Dat[4] := 16
    Dat[5] := 10
    Dat[6] := 24
    Dat[7] := 15
    Dat[8] := 15
    Dat[9] := 12
    Dat[10] := 8
    Dat[11] := 7
    Dat[12] := 6
    m := 0
    нц для k от 1 до 12
        если Dat[k] < 15 то
            m := m+1
        все
    кц
    вывод m
кон
```

Ответ: _____.

7

У исполнителя Уменьшитель две команды, которым присвоены номера:

1) вычти 1

2) раздели на 2

Первая из них уменьшает число на экране на 1, вторая – уменьшает его в 2 раза.

Составьте алгоритм получения из числа 85 числа 10, содержащий не более пяти команд.

В ответе запишите только номера команд.

Например, 22211 – это алгоритм

раздели на 2

раздели на 2

раздели на 2

вычти 1

вычти 1,

который преобразует число 64 в 6.

Если таких алгоритмов более одного, то запишите любой из них.

Ответ: _____.

8

Некоторый алгоритм из одной цепочки символов получает новую цепочку следующим образом. Сначала вычисляется длина исходной цепочки символов; если она чётна, то в середину цепочки символов добавляется символ **Ч**, а если нечётна, то в начало цепочки добавляется символ **Н**. В полученной цепочке символов каждая буква заменяется буквой, предшествующей ей в русском алфавите (**Б** – на **А**, **В** – на **Б** и т. д., а **А** – на **Я**).

Получившаяся таким образом цепочка является результатом работы алгоритма.

Например, если исходной цепочкой была цепочка **ВРМ**, то результатом работы алгоритма будет цепочка **МБПЛ**, а если исходной цепочкой была **ПД**, то результатом работы алгоритма будет цепочка **ОЦГ**.

Алгоритм применили дважды к некоторой исходной цепочке (то есть его применили к исходной цепочке, а затем вновь применили к результату). В итоге получили цепочку **ЛНЦЖР**. Определите исходную цепочку.

Русский алфавит: **АБВГДЕЁЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ**.

Ответ: _____.

Часть 2

Задание 9.1 или 9.2 выполняется на компьютере. Результатом исполнения задания является отдельный файл (для одного задания – один файл). Формат файла, его имя и каталог для сохранения Вам сообщит учитель.

Выберите для выполнения **ОДНО** из предложенных ниже заданий: **9.1** или **9.2**.

9.1

Исполнитель Робот умеет перемещаться по лабиринту, начерченному на плоскости, разбитой на клетки. Ниже приведено описание Робота.

У Робота есть четыре команды перемещения:

вверх

вниз

влево

вправо

При выполнении любой из этих команд Робот перемещается на одну клетку соответственно: вверх ↑, вниз ↓, влево ←, вправо →.

Между соседними (по сторонам) клетками может стоять стена, через которую Робот пройти не может. Если Робот получает команду передвижения через стену, то он разрушается.

Четыре команды проверяют истинность условия отсутствия стены у каждой стороны той клетки, где находится Робот:

сверху свободно

снизу свободно

слева свободно

справа свободно

Эти команды можно использовать вместе с условием **если**, имеющим следующий вид:

если <условие> то

последовательность команд

все

Последовательность команд – это одна или несколько любых команд, выполняемых Роботом. Например, для передвижения на одну клетку вправо, если справа нет стенки, можно использовать такой алгоритм:

если справа свободно то

вправо

все

В одном условии можно использовать несколько команд, применяя логические связки **и**, **или**, **не**, например:

если (справа свободно) и (не снизу свободно) то

вправо

все

Для повторения последовательности команд можно использовать цикл **пока**, имеющий следующий вид:

нц пока <условие>

последовательность команд

кц

Например, для безопасного движения вправо можно использовать следующий алгоритм:

нц пока справа свободно

вправо

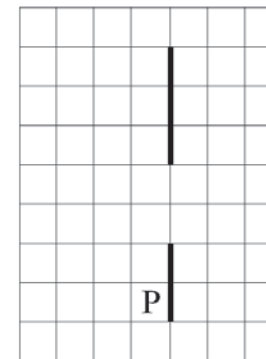
кц

Также у Робота есть команда **закрасить**, закрашивающая клетку, в которой Робот находится в настоящий момент.

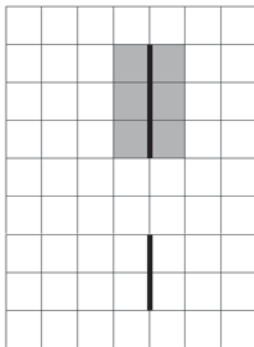
Выполните задание.

На бесконечном поле имеется вертикальная стена. **Длина стены неизвестна**. В стене есть ровно один проход, точное место прохода и его ширина неизвестны. Робот находится в клетке, расположенной непосредственно слева от стены у её нижнего конца.

На рисунке указан один из возможных способов расположения стены и Робота (Робот обозначен буквой «Р»):



Напишите для Робота алгоритм, закрашивающий все клетки, примыкающие вертикальными сторонами к участку стены выше прохода. Робот должен закрасить только клетки, удовлетворяющие данному условию. Например, для приведённого выше рисунка Робот должен закрасить следующие клетки (см. рисунок):



Конечное расположение Робота может быть произвольным. Алгоритм должен решать задачу для произвольного размера поля и любого допустимого расположения стен внутри прямоугольного поля. При исполнении алгоритма Робот не должен разрушиться. Алгоритм напишите в текстовом редакторе и сохраните в текстовом файле. Название файла и каталог для сохранения Вам сообщит учитель.

- 9.2** Напишите программу, которая в последовательности натуральных чисел определяет минимальное число, делящееся нацело на 7. Программа получает на вход количество чисел в последовательности, а затем сами числа. В последовательности всегда имеется число, делящееся нацело на 7. Количество чисел не превышает 1000. Введённые числа не превышают 30 000. Программа должна вывести одно число – минимальное число, делящееся нацело на 7.

Пример работы программы:

Входные данные	Выходные данные
4 21 100 4 490	21

Критерии оценивания заданий с развёрнутым ответом

9.1 Исполнитель Робот умеет перемещаться по лабиринту, начерченному на плоскости, разбитой на клетки. Ниже приведено описание Робота.

У Робота есть четыре команды перемещения:

вверх

вниз

влево

вправо

При выполнении любой из этих команд Робот перемещается на одну клетку соответственно: вверх ↑, вниз ↓, влево ←, вправо →.

Между соседними (по сторонам) клетками может стоять стена, через которую Робот пройти не может. Если Робот получает команду передвижения через стену, то он разрушается.

Четыре команды проверяют истинность условия отсутствия стены у каждой стороны той клетки, где находится Робот:

сверху свободно

снизу свободно

слева свободно

справа свободно

Эти команды можно использовать вместе с условием **если**, имеющим следующий вид:

если <условие> то

последовательность команд

все

Последовательность команд — это одна или несколько любых команд, выполняемых Роботом. Например, для передвижения на одну клетку вправо, если справа нет стенки, можно использовать такой алгоритм:

если справа свободно то

вправо

все

В одном условии можно использовать несколько команд, применяя логические связки **и**, **или**, **не**, например:

если (справа свободно) и (не снизу свободно) то

вправо

все

Для повторения последовательности команд можно использовать цикл **пока**, имеющий следующий вид:

нц пока <условие>

последовательность команд

кц

Например, для безопасного движения вправо можно использовать следующий алгоритм:

нц пока справа свободно

вправо

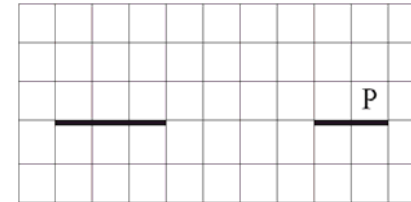
кц

Также у Робота есть команда **закрасить**, закрашивающая клетку, в которой Робот находится в настоящий момент.

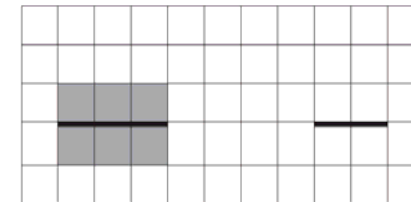
Выполните задание.

На бесконечном поле имеется горизонтальная стена. **Длина стены неизвестна**. В стене есть ровно один проход, точное место прохода и его ширина неизвестны. Робот находится в клетке, расположенной непосредственно над стеной у её правого конца.

На рисунке указан один из возможных способов расположения стен и Робота (Робот обозначен буквой «Р»):



Напишите для Робота алгоритм, закрашивающий все клетки, примыкающие горизонтальными сторонами к участку стены левее прохода. Робот должен закрасить только клетки, удовлетворяющие данному условию. Например, для приведённого выше рисунка Робот должен закрасить следующие клетки (см. рисунок):



Конечное расположение Робота может быть произвольным. Алгоритм должен решать задачу для произвольного размера поля и любого допустимого расположения стен внутри прямоугольного поля. При исполнении алгоритма Робот не должен разрушиться.

Алгоритм напишите в текстовом редакторе и сохраните в текстовом файле.

Название файла и каталог для сохранения Вам сообщит учитель.

Содержание верного ответа и указания по оцениванию (допускаются иные формулировки ответа, не искажающие его смысла)	
Команды исполнителя будем записывать жирным шрифтом, а комментарии, поясняющие алгоритм и не являющиеся его частью, – курсивом. Начало комментария будем обозначать символом « ». Двигаемся влево, пока не дойдём до прохода нц пока не снизу свободно влево кц Двигаемся влево до конца прохода нц пока снизу свободно влево кц Двигаемся влево, пока не дойдём до конца стены, закрашивая все клетки на пути нц пока не снизу свободно закрасить влево кц Переходим на нижнюю сторону стены вниз вправо Двигаемся вправо, пока не дойдём до конца стены, закрашивая все клетки на пути нц пока не сверху свободно закрасить вправо кц Возможны и другие варианты решения.	
Указания по оцениванию	Баллы
Алгоритм правильно работает при всех допустимых исходных данных	2
При всех допустимых исходных данных верно следующее: 1) выполнение алгоритма завершается, и при этом Робот не разбивается; 2) закрашено не более 5 лишних клеток; 3) остались не закрашенными не более 5 клеток из числа тех, которые должны были быть закрашены	1
Задание выполнено неверно, т. е. не выполнены условия, позволяющие поставить 1 или 2 балла	0
Максимальный балл	2

9.2

Напишите программу, которая в последовательности натуральных чисел определяет максимальное число, оканчивающееся на 8. Программа получает на вход количество чисел в последовательности, а затем сами числа. В последовательности всегда имеется число, оканчивающееся на 8.

Количество чисел не превышает 1000. Введённые числа не превышают 30 000.

Программа должна вывести одно число – максимальное число, оканчивающееся на 8.

Пример работы программы:

Входные данные	Выходные данные
4 138 81 80 108	138

Содержание верного ответа и указания по оцениванию (допускаются иные формулировки ответа, не искажающие его смысла)		
Решением является программа, записанная на любом языке программирования. Пример верного решения, записанного на языке Паскаль: <pre>var n,i,a, max: integer; begin max:=0; readln(n); for i:=1 to n do begin readln(a); if (a mod 10 = 8) and (a > max) then max:=a; end; writeln(max) end.</pre>		
Возможны и другие варианты решения.		
Для проверки правильности работы программы необходимо использовать следующие тесты:		
№	Входные данные	Выходные данные
1	3 18 138 120	138
2	4 88 128 48 273	128
3	1 8	8

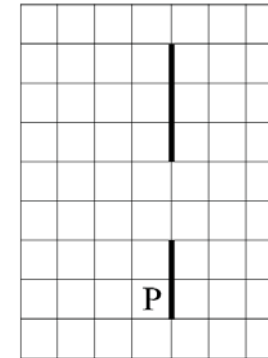
Указания по оцениванию	Баллы
Предложено верное решение. Программа правильно работает на всех приведённых выше тестах. Программа может быть записана на любом языке программирования	2
Программа выдаёт неверный ответ ровно на одном из тестов, приведённых выше. Например, решение, в котором проверяется делимость на 8 вместо проверки остатка, выдаст неправильный ответ на тесте № 1	1
Программа выдаёт на тестах неверные ответы, отличные от описанных в критерии на 1 балл	0
Максимальный балл	2

Критерии оценивания заданий с развёрнутым ответом

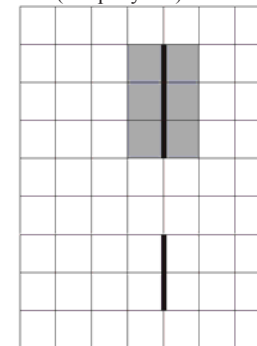
- 9.1** Исполнитель Робот умеет перемещаться по лабиринту, начерченному на плоскости, разбитой на клетки. Ниже приведено описание Робота.
У Робота есть четыре команды перемещения:
вверх
вниз
влево
вправо
При выполнении любой из этих команд Робот перемещается на одну клетку соответственно: вверх ↑, вниз ↓, влево ←, вправо →.
Между соседними (по сторонам) клетками может стоять стена, через которую Робот пройти не может. Если Робот получает команду передвижения через стену, то он разрушается.
Четыре команды проверяют истинность условия отсутствия стены у каждой стороны той клетки, где находится Робот:
сверху свободно
снизу свободно
слева свободно
справа свободно
Эти команды можно использовать вместе с условием **если**, имеющим следующий вид:
если <условие> то
последовательность команд
все
Последовательность команд – это одна или несколько любых команд, выполняемых Роботом. Например, для передвижения на одну клетку вправо, если справа нет стенки, можно использовать такой алгоритм:
если справа свободно то
вправо
все
В одном условии можно использовать несколько команд, применяя логические связки **и**, **или**, **не**, например:
если (справа свободно) и (не снизу свободно) то
вправо
все
Для повторения последовательности команд можно использовать цикл **пока**, имеющий следующий вид:
нц пока <условие>
последовательность команд
кц
Например, для безопасного движения вправо можно использовать следующий алгоритм:
нц пока справа свободно
вправо
кц
Также у Робота есть команда **закрасить**, закрашивающая клетку, в которой Робот находится в настоящий момент.

Выполните задание.

На бесконечном поле имеется вертикальная стена. **Длина стены неизвестна**. В стене есть ровно один проход, точное место прохода и его ширина неизвестны. Робот находится в клетке, расположенной непосредственно слева от стены у её нижнего конца.
На рисунке указан один из возможных способов расположения стен и Робота (Робот обозначен буквой «Р»):



Напишите для Робота алгоритм, закрашивающий все клетки, примыкающие вертикальными сторонами к участку стены выше прохода. Робот должен закрасить только клетки, удовлетворяющие данному условию. Например, для приведённого выше рисунка Робот должен закрасить следующие клетки (см. рисунок):



Конечное расположение Робота может быть произвольным. Алгоритм должен решать задачу для произвольного размера поля и любого допустимого расположения стен внутри прямоугольного поля. При выполнении алгоритма Робот не должен разрушиться. Алгоритм напишите в текстовом редакторе и сохраните в текстовом файле. Название файла и каталог для сохранения Вам сообщит учитель.

Содержание верного ответа и указания по оцениванию (допускаются иные формулировки ответа, не искажающие его смысла)	
Команды исполнителя будем записывать жирным шрифтом, а комментарии, поясняющие алгоритм и не являющиеся его частью, – курсивом. Начало комментария будем обозначать символом « ». <i> Двигаемся вверх, пока не дойдём до прохода</i> нц пока не справа свободно вверх кц <i> Двигаемся вверх до конца прохода</i> нц пока справа свободно вверх кц <i> Двигаемся вверх, пока не дойдём до конца стены, закрашивая все клетки на пути</i> нц пока не справа свободно закрасить вверх кц <i> Переходим на другую сторону стены</i> вправо вниз <i> Двигаемся вниз, пока не дойдём до конца стены, закрашивая все клетки на пути</i> нц пока не слева свободно закрасить вниз кц Возможны и другие варианты решения.	
Указания по оцениванию	Баллы
Алгоритм правильно работает при всех допустимых исходных данных	2
При всех допустимых исходных данных верно следующее: 1) выполнение алгоритма завершается, и при этом Робот не разбивается; 2) закрашено не более 5 лишних клеток; 3) остались не закрашенными не более 5 клеток из числа тех, которые должны были быть закрашены	1
Задание выполнено неверно, т. е. не выполнены условия, позволяющие поставить 1 или 2 балла	0
<i>Максимальный балл</i>	2

9.2

Напишите программу, которая в последовательности натуральных чисел определяет минимальное число, делящееся нацело на 7. Программа получает на вход количество чисел в последовательности, а затем сами числа. В последовательности всегда имеется число, делящееся нацело на 7.

Количество чисел не превышает 1000. Введённые числа не превышают 30 000.

Программа должна вывести одно число – минимальное число, делящееся нацело на 7.

Пример работы программы:

Входные данные	Выходные данные
4 21 100 4 490	21

Содержание верного ответа и указания по оцениванию (допускаются иные формулировки ответа, не искажающие его смысла)		
Решением является программа, записанная на любом языке программирования. Пример верного решения, записанного на языке Паскаль: <pre>var n,i,a, min: integer; begin min:=30001; readln(n); for i:=1 to n do begin readln(a); if (a mod 7 = 0) and (a < min) then min:=a; end; writeln(min) end.</pre>		
Возможны и другие варианты решения.		
Для проверки правильности работы программы необходимо использовать следующие тесты:		
№	Входные данные	Выходные данные
1	3 17 28 56	28
2	4 567 77 497 18	77
3	1 7	7

Указания по оцениванию	Баллы
Предложено верное решение. Программа правильно работает на всех приведённых выше тестах. Программа может быть записана на любом языке программирования	2
Программа выдаёт неверный ответ ровно на одном из тестов, приведённых выше. Например, решение, в котором вместо делимости на 7 проверяется равенство последней цифры 7, выдаст неправильный ответ на тесте № 1	1
Программа выдаёт на тестах неверные ответы, отличные от описанных в критерии на 1 балл	0
Максимальный балл	2

**Спецификация
контрольных измерительных материалов
для проведения диагностических тематических работ
по подготовке к ОГЭ
по ИНФОРМАТИКЕ И ИКТ**

1. Назначение КИМ – оценить уровень общеобразовательной подготовки по информатике и ИКТ обучающихся основной школы. Диагностические тематические работы предназначены для контроля освоения крупных содержательных разделов курса (рубежного контроля), а также могут использоваться в период предэкзаменационной подготовки. Выполнение полной серии тематических диагностических работ дает представление о достижении обучающимися требований к уровню подготовки выпускников и готовности к успешной сдаче ОГЭ по информатике и ИКТ.

Серия включает шесть тематических работ.

Таблица 1. Перечень диагностических работ

Класс	№ п/п	Название работы	Тематические блоки	Продолжительность
8 класс	1	Файловая система персонального компьютера.	Основные устройства ИКТ Организация информационной среды, поиск информации Представление и передача информации	45 мин
	2	Компьютерное представление и обработка текстовой информации.	Запись средствами ИКТ информации об объектах и о процессах, создание и обработка информационных объектов Организация информационной среды, поиск информации Основные устройства ИКТ Представление и передача информации	45 мин
	3	Простые вычисления с помощью электронных таблиц.	Математические инструменты, электронные таблицы	45 мин
9 класс	4	Представление информации в виде схем и таблиц. Передача информации и информационный поиск.	Представление и передача информации Организация информационной среды, поиск информации	45 мин
	5	Логика и алгоритмы	Представление и передача информации Обработка информации	90 мин
	6	Обработка таблиц: выбор и сортировка записей.	Запись средствами ИКТ информации об объектах и о процессах, создание и обработка информационных объектов	45 мин

Для классификации элементов содержания и требований к уровню подготовки учащихся общеобразовательных учреждений используется кодификатор ОГЭ по информатике и ИКТ.

2. Документы, определяющие содержание КИМ

Содержание тематических работ определяет Федеральный компонент государственного стандарта основного общего образования по информатике и ИКТ (приказ Минобрнауки России от 05.03.2004 № 1089 «Об утверждении федерального компонента государственных стандартов начального общего, основного общего и среднего (полного) общего образования»).

Содержание работ соответствует Федеральному государственному образовательному стандарту основного общего образования (Приказ Минобрнауки РФ от 17 декабря 2010 г. № 1897).

3. Подходы к отбору содержания, разработке структуры КИМ

Диагностические работы охватывают основное содержание курса информатики и ИКТ. Охвачен наиболее значимый материал, однозначно трактуемый в большинстве преподаваемых в школе вариантов курса информатики и ИКТ и учебниках Федерального перечня Минобрнауки.

Содержание заданий разработано по основным темам курса информатики и ИКТ, объединенных в следующие тематические блоки: «Представление и передача информации» (разделы 1.1 и 1.2 кодификатора), «Обработка информации» (разделы 1.3 и 1.4 кодификатора), «Основные устройства ИКТ» (раздел 2.1 кодификатора), «Запись средствами ИКТ информации об объектах и о процессах, создание и обработка информационных объектов» (разделы 2.2 и 2.3 кодификатора), «Проектирование и моделирование» (раздел 2.5 кодификатора), «Математические инструменты, электронные таблицы» (раздел 2.6 кодификатора), «Организация информационной среды, поиск информации» (разделы 2.7 и 2.4 кодификатора).

В работы не включены задания, требующие простого воспроизведения знания терминов, понятий, величин, правил (такие задания слишком просты для выполнения). При выполнении любого из заданий от учащегося требуется решить какую-либо задачу: либо прямо использовать известное правило, алгоритм, умение; либо выбрать из общего количества изученных понятий и алгоритмов наиболее подходящее и применить его в известной либо новой ситуации.

Часть 2 всех работ состоит из практических заданий, проверяющих наиболее важные практические навыки курса информатики и ИКТ: умение выполнить информационный поиск, обработать большой массив числовых данных и умение разработать и записать простой алгоритм.

Практические задания не требуют от обучающихся знания конкретных операционных систем и программных продуктов, навыков работы с ними. Проверяемыми элементами являются основные принципы представления, хранения и обработки информации, навыки работы с такими категориями программного обеспечения, как операционная система, текстовый редактор, электронная (динамическая) таблица и среда формального исполнителя, а не знание особенностей конкретных программных продуктов. Практические задания могут быть выполнены с использованием различных операционных систем и прикладных программных продуктов.

4. Характеристика структуры и содержания КИМ

Каждая диагностическая работа состоит из двух частей. В части 1 сгруппированы различные задания с кратким ответом, для выполнения которых не нужен компьютер. Задания части 2 подразумевают практическую работу обучающихся за компьютером с использованием специального программного обеспечения. Результатом исполнения каждого практического задания может быть краткий ответ или развернутый ответ в форме отдельного файла, проверяемый учителем (экспертом) в соответствии с критериями. Для выполнения большинства заданий части 2 требуется исходный файл или каталог файлов, который является частью контрольных измерительных материалов

Распределение заданий по частям диагностических работ представлено в таблице 2.

Таблица 2. Распределение заданий по частям диагностических работ

№ рабо- ты	Часть 1			Часть 2			Всего заданий в ра- боте	Максимальный пер- вичный балл за всю работу
	Чис ло зада дан- ний	Макс. пер- вич- ный балл	Процент макс. первичного бал- ла за задания данной части от макс. первичного балла за всю ра- боту	Число зада- ний	Макс. пер- вич- ный балл	Процент макс. первичного бал- ла за задания данной части от макс. первичного балла за всю ра- боту		
8 класс								
1	5	5	56	4	4	44	9	9
2	5	5	56	3	4	44	8	9
3	5	5	63	3	3	37	8	8
9 класс								
4	6	6	75	2	2	25	8	8
5	8	8	80	1	2	20	9	10
6	4	4	67	2	2	33	6	6

5. Распределение заданий КИМ по содержанию и видам деятельности

В работу включены задания из всех разделов, изучаемых в курсе информатики и ИКТ. Распределение заданий по разделам курса информатики представлено в таблице 3.

Таблица 3. Распределение заданий по разделам курса информатики

№ п/п	Название раздела	Число заданий
1.	Представление и передача информации	8 класс Работа №1 – 1; Работа №2 – 1 9 класс Работа №4 – 2; Работа №5 – 2
2.	Обработка информации	9 класс Работа №5 – 7
3.	Основные устройства ИКТ	8 класс Работа №1 – 4; Работа №2 – 2
4.	Запись средствами ИКТ информации об объектах и о процессах, создание и обработка информационных объектов	8 класс Работа №2 – 4 9 класс

		Работа №6 – 6
5.	Проектирование и моделирование	9 класс Работа №4 – 1
6.	Математические инструменты, электронные таблицы	8 класс Работа №3 – 8
7.	Организация информационной среды, поиск информации	8 класс Работа №1 – 4; Работа №2 – 1 9 класс Работа №4 – 5

На уровне воспроизведения знаний проверяется такой фундаментальный теоретический материал, как:

- единицы измерения информации;
- принципы кодирования информации;
- моделирование;
- понятие алгоритма, его свойств, способов записи;
- основные алгоритмические конструкции (ветвление и циклы);
- основные элементы математической логики;
- основные понятия, используемые в информационных и коммуникационных технологиях;
- принципы организации файловой системы.

Задания, проверяющие сформированность умений применять свои знания в стандартной ситуации, также включены в часть 1 работ. Это следующие умения:

- подсчитывать информационный объем сообщения;
- использовать стандартные алгоритмические конструкции для построения алгоритмов для формальных исполнителей;
- формально исполнять алгоритмы, записанные на естественном и алгоритмическом языках;
- оценивать результат работы известного программного обеспечения;
- формулировать запросы к базам данных и поисковым системам.

Материал на проверку сформированности умений применять свои знания в новой ситуации входит в часть 2 работ. Это следующие сложные умения:

- выполнение информационного поиска средствами операционной системы или текстового редактора с самостоятельным формулированием запросов;
- применение технологии обработки текста с использованием средств текстового редактора;
- разработка технологии обработки информационного массива с использованием средств электронной таблицы или базы данных;
- разработка алгоритма для формального исполнителя или на языке программирования с использованием условных инструкций и циклов, а также логических связок при задании условий.

6. Продолжительность выполнения диагностической работы

На выполнение диагностических работ №1-4 и 6 отводится 45 минут, работы № 5 – 90 минут.

7. Дополнительные материалы и оборудование

Задания части 1 выполняются обучающимися без использования компьютеров и других технических средств. Вычислительная сложность заданий не требует использования калькуляторов, поэтому в целях обеспечения равенства всех выполняющих работу использование калькуляторов не разрешается.

Задания части 2 выполняются обучающимися на компьютере. На компьютере должны быть установлены знакомые обучающимся программы. Для выполнения практических заданий, связанных с обработкой массивов числовых данных, на компьютере должен быть установлен редактор электронных таблиц. Для выполнения практических заданий, связанных с обработкой текстов, на компьютере должен быть установлен текстовый редактор.

Практическое задание по разработке алгоритмов (программ) в работе №2 (9-й класс) дается в двух вариантах по выбору учащегося. Первый вариант задания предусматривает разработку алгоритма для исполнителя «Робот». Для выполнения этого варианта задания рекомендуется использование учебной среды исполнителя «Робот». В качестве такой среды может использоваться, например, учебная среда разработки «Кумир» (<http://www.niisi.ru/kumir>), или любая другая среда, позволяющая моделировать исполнителя «Робот». В случае, если синтаксис команд исполнителя в используемой среде отличается от того, который дан в задании, допускается внесение изменений в текст задания в части описания исполнителя «Робот». При отсутствии учебной среды исполнителя «Робот» решение этого задания записывается в простом текстовом редакторе.

Второй вариант задания предусматривает запись алгоритма на изучаемом языке программирования (если изучение темы «Алгоритмизация» проводится с использованием языка программирования). В этом случае для выполнения задания необходима система программирования, используемая при обучении.

8. Система оценивания выполнения заданий и работы в целом

Правильное выполнение каждого задания с кратким ответом оценивается в 1 балл. За выполнение каждого задания присваивается (в дихотомической системе оценивания) либо 0 баллов («задание не выполнено»), либо 1 балл («задание выполнено»).

Выполнение заданий с развернутым ответом в виде компьютерного файла (задание 8 работы №2 для 8 класса и задание 9 работы №2 для 9 класса) оценивается от 0 до 2 баллов. Ответы на эти задания проверяются и оцениваются по определенному перечню критериев.

Таблица 4. Таблица перевода баллов в отметки по пятибалльной шкале

Отметка по пятибалльной шкале	«2»	«3»	«4»	«5»
Работы 1 и 2	0-2	3-5	6-7	8-9
Работы 3 и 4	0-2	3-4	5-6	7-8
Работа 5	0-2	3-5	6-8	9-10
Работа 6	0-1	2-3	4-5	6

**Обобщенный план варианта
контрольных измерительных материалов
для проведения диагностических тематических работ
по подготовке к ГИА-9**

Уровни сложности заданий: Б – базовый (примерный интервал выполнения задания – 60–90%);
П – повышенный (40–60%); В – высокий (менее 40%).

Работа №1

№	Проверяемые элементы содержания	Коды про- веряемых элементов содержания по кодифи- катору	Коды тре- бований к уровню подготовки по кодифи- катору	Уро- вень слож- ности зада- ния	Макс. балл за выпол- нение задания	Пример- ное время выполне- ния зада- ния (мин.)
Часть 1						
1	Файлы и файловая система	2.1.2	1.5	Б	1	3
2	Файлы и файловая система	2.1.2	1.5	Б	1	3
3	Файлы и файловая система	2.1.2	1.5	Б	1	4
4	Единицы измерения количества ин- формации	1.1.3	1.2	Б	1	3
5	Оценка количественных параметров информационных объектов. Объем памяти, необходимый для хранения объектов	2.1.3	2.3	П	1	6
Часть 2						
6	Компьютерные и некомпьютерные каталоги, поисковые машины, форму- лирование запросов	2.4.1	2.1	П	1	5
7	Компьютерные и некомпьютерные каталоги, поисковые машины, форму- лирование запросов	2.4.1	2.1	П	1	5
8	Компьютерные и некомпьютерные каталоги, поисковые машины, форму- лирование запросов	2.4.1	2.1	П	1	6
9	Компьютерные и некомпьютерные каталоги, поисковые машины, форму- лирование запросов	2.4.1	2.1	В	1	10
Всего заданий – 9; из них по уровню сложности: Б – 4, П – 4, В – 1. Максимальный балл – 9. Общее время выполнения работы – 45 мин.						

Работа №2

№	Проверяемые элементы содержания	Коды про- веря- емых элемен- тов со- держания по коди- фикатору	Коды тре- бова- ний к уровню подго- товки по кодифи- катору	Уро- вень слож- ности зада- ния	Макс. балл за вы- полне- ние зада- ния	Пример- ное время выполне- ния зада- ния (мин.)
Часть 1						
1	Запись текстовой информации с ис- пользованием различных устройств	2.2.2	2.4.1	Б	1	2
2	Работа с фрагментами текста. Страни- ца. Абзацы, ссылки, заголовки, оглав- ления.	2.3.1	2.4.1	Б	1	3
3	Проверка правописания, словари.	2.3.1	2.4.1	Б	1	3
4	Дискретная форма представления ин- формации.	1.1.3	1.2	Б	1	3
5	Оценка количественных параметров информационных объектов. Объем памяти, необходимый для хранения объектов	2.1.3	2.3	П	1	4
Часть 2						
6	Компьютерные и некомпьютерные ка- талоги, поисковые машины, формули- рование запросов	2.4.1	2.1	П	1	5
7	Оценка количественных параметров информационных объектов.	2.1.3	1.5	П	1	5
8	Создание текста посредством квали- фицированного клавиатурного письма с использованием базовых средств тек- стовых редакторов. Включение в текст списков, таблиц, изображений, диаграмм, формул	2.3.1	2.4.1	В	2	20
Всего заданий – 8; из них по уровню сложности: Б – 4, П – 3, В – 1. Максимальный балл – 9. Общее время выполнения работы – 45 мин.						

Работа №3

№	Проверяемые элементы содержания	Коды про- веряемых элементов содержания по кодифи- катору	Коды тре- бований к уровню подготовки по кодифи- катору	Уро- вень слож- ности зада- ния	Макс. балл за выпол- нение задания	Пример- ное время выполне- ния зада- ния (мин.)
Часть 1						
1	Таблица как средство моделирования.	2.6.1	2.4.2	Б	1	3
2	Таблица как средство моделирования.	2.6.1	2.4.2	Б	1	3
3	Таблица как средство моделирования.	2.6.1	2.4.2	Б	1	3
4	Умение представлять формульную зависимость в графическом виде	2.6.3	2.4.2	Б	1	3
5	Умение представлять формульную зависимость в графическом виде	2.6.3	2.4.2	П	1	4
Часть 2						
6	Ввод математических формул и вычисления по ним	2.6.2	2.4.2	П	1	6
7	Ввод математических формул и вычисления по ним	2.6.2	2.4.2	П	1	8
8	Ввод математических формул и вычисления по ним	2.6.2	3.1	В	1	15
Всего заданий – 8 ; из них по уровню сложности: Б – 4 , П – 3 , В – 1 . Максимальный балл – 8 . Общее время выполнения работы – 45 мин.						

9 класс Работа №4

№	Проверяемые элементы содержания	Коды про- веряемых элементов содержания по кодифи- катору	Коды тре- бований к уровню подготовки по кодифи- катору	Уро- вень слож- ности зада- ния	Макс. балл за выпол- нение задания	Пример- ное время выполне- ния зада- ния (мин.)
Часть 1						
1	Формализация описания реальных объектов и процессов, моделирование объектов и процессов	1.1.2	2.4.2	Б	1	4
2	Формализация описания реальных объектов и процессов, моделирование объектов и процессов	1.1.2	2.4.2	Б	1	4
3	Умение анализировать информацию, представленную в виде схем	2.5.2	2.4.2	Б	1	4
4	Оценка количественных параметров информационных процессов. Скорость передачи и обработки объектов.	2.1.4	1.2	Б	1	3
5	Сохранение информационных объектов из компьютерных сетей и ссылок на них	2.7.3	3.4	П	1	5
6	Информация в компьютерных сетях. Поисковые машины	2.4.1	2.5	П	1	5
Часть 2						
7	Компьютерные и некомпьютерные каталоги, формулирование запросов	2.4.1	2.5	П	1	8
8	Компьютерные и некомпьютерные каталоги, формулирование запросов	2.4.1	2.5	В	1	12
Всего заданий – 8 ; из них по уровню сложности: Б – 4 , П – 3 , В – 1 . Максимальный балл – 8 . Общее время выполнения работы – 45 мин.						

Работа №5

№	Проверяемые элементы содержания	Коды проверяемых элементов содержания по кодификатору	Коды требований к уровню подготовки по кодификатору	Уровень сложности задания	Макс. балл за выполнение задания	Примерное время выполнения задания (мин.)
Часть 1						
1	Логические значения, операции, выражения	1.3.3	2.1	Б	1	4
2	Кодирование и декодирование информации	1.2.2	2.1	Б	1	4
3	Алгоритм, свойства алгоритмов, способы записи алгоритмов.	1.3.1	2.1	Б	1	4
4	Алгоритм, свойства алгоритмов, способы записи алгоритмов.	1.3.1	2.1	Б	1	4
5	Алгоритмические конструкции.	1.3.2	2.1	Б	1	5
6	Обрабатываемые объекты: числа, списки.	1.3.5	2.1	П	1	8
7	Алгоритм, свойства алгоритмов, способы записи алгоритмов.	1.3.1	2.1	П	1	8
8	Кодирование и декодирование информации	1.2.2	2.1	П	1	8
Часть 2						
9	Умение написать короткий алгоритм в среде формального исполнителя (вариант задания 9.1) или на языке программирования (вариант задания 9.2)	1.3	3.1	В	2	45
Всего заданий – 9 ; из них по уровню сложности: Б – 5 , П – 3 , В – 1 . Максимальный балл – 10 . Общее время выполнения работы – 90 мин.						

Работа №6

№	Проверяемые элементы содержания	Коды проверяемых элементов содержания по кодификатору	Коды требований к уровню подготовки по кодификатору	Уровень сложности задания	Макс. балл за выполнение задания	Примерное время выполнения задания (мин.)
Часть 1						
1	Базы данных. Поиск данных в готовой базе.	2.3.2	2.5	Б	1	3
2	Базы данных. Поиск данных в готовой базе.	2.3.2	2.5	Б	1	3
3	Базы данных. Поиск данных в готовой базе.	2.3.2	2.5	П	1	5
4	Базы данных. Поиск данных в готовой базе.	2.3.2	2.5	П	1	6
Часть 2						
5	Базы данных. Поиск данных в готовой базе.	2.3.2	2.5	В	1	14
6	Базы данных. Поиск данных в готовой базе.	2.3.2	2.5	В	1	14
Всего заданий – 6 ; из них по уровню сложности: Б – 2 , П – 2 , В – 2 . Максимальный балл – 6 . Общее время выполнения работы – 45 мин.						

Диагностическая тематическая работа № 5

**по подготовке к ОГЭ
по ИНФОРМАТИКЕ и ИКТ**

по теме «Логика и алгоритмы»

Инструкция по выполнению работы

На выполнение диагностической работы по информатике отводится 90 минут. Работа включает в себя 9 заданий. Задание 9.1 или 9.2 выполняется по выбору обучающегося.

Ответы к заданиям 1 и 3 записываются в виде одной цифры, которая соответствует номеру правильного ответа. Эту цифру запишите в поле ответа в тексте работы.

Ответы к заданиям 2, 4–8 записываются в виде последовательности букв, числа в поле ответа в тексте работы. В случае записи неверного ответа зачеркните его и запишите рядом новый.

Задания 9.1 или 9.2 выполняются на компьютере. Результатом выполнения задания 9.1 или 9.2 является файл, который необходимо сохранить под именем и в формате, указанном учителем.

Советуем выполнять задания в том порядке, в котором они даны. В целях экономии времени пропускайте задание, которое не удаётся выполнить сразу, и переходите к следующему. Если после выполнения всей работы у Вас останется время, то Вы сможете вернуться к пропущенным заданиям. Постарайтесь выполнить как можно больше заданий.

Желаем успеха!

Часть 1**1**

Для какого из приведённых имён истинно высказывание:
НЕ(Первая буква гласная) И НЕ(Последняя буква согласная)?

- 1) Иван
- 2) Михаил
- 3) Семён
- 4) Никита

Ответ:

2

Ваня шифрует русские слова, записывая вместо каждой буквы её номер в алфавите (без пробелов). Номера букв даны в таблице.

А	1	Й	11	У	21	Э	31
Б	2	К	12	Ф	22	Ю	32
В	3	Л	13	Х	23	Я	33
Г	4	М	14	Ц	24		
Д	5	Н	15	Ч	25		
Е	6	О	16	Ш	26		
Ё	7	П	17	Щ	27		
Ж	8	Р	18	Ъ	28		
З	9	С	19	Ы	29		
И	10	Т	20	Ь	30		

Некоторые шифровки можно расшифровать не одним способом. Например, 311333 может означать «ВАЛЯ», может – «ЭЛЯ», или «ВААВВВ».

Вот четыре шифровки:

3113
9212
6810
2641

Только одна из них расшифровывается единственным способом. Найдите её и расшифруйте. То, что получилось, запишите в качестве ответа.

Ответ: _____.

3

Исполнитель Чертёжник перемещается на координатной плоскости, оставляя след в виде линии. Чертёжник может выполнять команду **Сместиться на (a, b)** (где a, b – целые числа), перемещающую Чертёжника из точки с координатами (x, y) в точку с координатами $(x + a; y + b)$. Если числа a, b положительные, значение соответствующей координаты увеличивается, если отрицательные, то уменьшается.

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

Запись

Повтори k раз

Команда1 Команда2 Команда3

конец

означает, что последовательность команд **Команда1 Команда2 Команда3** повторится k раз.

Чертёжнику был дан для исполнения следующий алгоритм:

Повтори 3 раз

Сместиться на (–3, –2) Сместиться на (2, 1) Сместиться на (3, 0)

конец

Какую команду надо выполнить Чертёжнику, чтобы вернуться в исходную точку, из которой он начал движение?

- 1) Сместиться на $(–3, –6)$
- 2) Сместиться на $(–6, 3)$
- 3) Сместиться на $(6, –3)$
- 4) Сместиться на $(3, 6)$

Ответ:

4

В алгоритме, записанном ниже, используются целочисленные переменные **a** и **b**, а также следующие операции.

Обозначение	Тип операции
$:=$	Присваивание
$+$	Сложение
$-$	Вычитание
$*$	Умножение
$/$	Деление

Определите значение переменной **a** после исполнения данного алгоритма.

```

a := 3
b := 2
b := 9 + a * b
a := b / 5 * a

```

Порядок действий соответствует правилам арифметики.

В ответе укажите одно число – значение переменной **a**.

Ответ: _____.

5

Запишите значение переменной **s**, полученное в результате работы следующей программы. Текст программы приведён на трёх языках программирования.

Алгоритмический язык	Бейсик	Паскаль
<pre> алг нач цел s, k s := 8 нц для k от 3 до 8 s := s + 8 кц вывод s кон </pre>	<pre> DIM k, s AS INTEGER s = 8 FOR k = 3 TO 8 s = s + 8 NEXT k PRINT s </pre>	<pre> Var s,k: integer; Begin s := 8; for k := 3 to 8 do s := s + 8; writeln(s); End. </pre>

Ответ: _____.

6

В таблице Dat хранятся данные о численности учеников в классах (Dat[1] – число учеников в первом классе; Dat[2] – во втором и т.д.). Определите, какое число будет напечатано в результате работы следующей программы.

Текст программы приведён на трёх языках программирования.

Паскаль	Бейсик
<pre> Var k, m: integer; Dat: array[1..11] of integer; Begin Dat[1] := 20; Dat[2] := 25; Dat[3] := 19; Dat[4] := 25; Dat[5] := 26; Dat[6] := 22; Dat[7] := 24; Dat[8] := 28; Dat[9] := 26; Dat[10] := 23; Dat[11] := 27; m := 0; for k := 1 to 11 do if Dat[k] > 22 then begin m := m + 1 end; writeln(m) end. </pre>	<pre> DIM Dat(11) AS INTEGER DIM k,m AS INTEGER Dat(1) = 20: Dat(2) = 25 Dat(3) = 19: Dat(4) = 25 Dat(5) = 26: Dat(6) = 22 Dat(7) = 24: Dat(8) = 28 Dat(9) = 26: Dat(10) = 23 Dat(11) = 27 m = 0 FOR k = 1 TO 11 IF Dat(k) > 22 THEN m = m + 1 END IF NEXT k PRINT m </pre>
Алгоритмический язык	
<pre> алг нач целтаб Dat[1:11] цел k, m Dat[1] := 20; Dat[2] := 25 Dat[3] := 19; Dat[4] := 25 Dat[5] := 26; Dat[6] := 22 Dat[7] := 24; Dat[8] := 28 Dat[9] := 26; Dat[10] := 23 Dat[11] := 27 m := 0 нц для k от 1 до 11 если Dat[k] > 22 то m := m + 1 все кц вывод m кон </pre>	

Ответ: _____.

7

У исполнителя Вычислитель две команды, которым присвоены номера:

1. **умножь на 3**

2. **вычти 2**

Первая из них увеличивает число на экране в 3 раза, вторая уменьшает его на 2.

Составьте алгоритм получения из числа 2 числа 30, содержащий не более пяти команд.

В ответе запишите только номера команд.

(Например, **11221** – это алгоритм

умножь на 3

умножь на 3

вычти 2

вычти 2

умножь на 3,

который преобразует число 1 в число 15.)

Если таких алгоритмов более одного, то запишите любой из них.

Ответ: _____.

8

Некоторый алгоритм из одной цепочки символов получает новую цепочку следующим образом. Сначала вычисляется длина исходной цепочки символов; если она чётна, то в середину цепочки символов добавляется символ **А**, а если нечётна, то в начало цепочки добавляется символ **Б**. В полученной цепочке символов каждая буква заменяется буквой, следующей за ней в русском алфавите (**А** – на **Б**, **Б** – на **В** и т.д., а **Я** – на **А**).

Получившаяся таким образом цепочка является результатом работы алгоритма.

Например, если исходной цепочкой была цепочка **ВРМ**, то результатом работы алгоритма будет цепочка **ВГСН**, а если исходной цепочкой была **ПД**, то результатом работы алгоритма будет цепочка **РБЕ**.

Дана цепочка символов **ПУСК**. Какая цепочка символов получится, если к данной цепочке применить описанный алгоритм дважды (т.е. применить алгоритм к данной цепочке, а затем к результату вновь применить алгоритм)?

Русский алфавит: **АБВГДЕЁЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ**.

Ответ: _____.

Часть 2

Задание 9.1 или 9.2 выполняется на компьютере. Результатом исполнения задания является отдельный файл (для одного задания – один файл). Формат файла, его имя и каталог для сохранения Вам сообщит учитель.

Выберите для выполнения **ОДНО** из предложенных ниже заданий: 9.1 или 9.2.

9.1

Исполнитель Робот умеет перемещаться по лабиринту, начерченному на плоскости, разбитой на клетки. Ниже приведено описание Робота.

У Робота есть четыре команды перемещения:

вверх

вниз

влево

вправо

При выполнении любой из этих команд Робот перемещается на одну клетку соответственно: вверх ↑, вниз ↓, влево ←, вправо →.

Между соседними (по сторонам) клетками может стоять стена, через которую Робот пройти не может. Если Робот получит команду передвижения через стену, то он разрушится.

Четыре команды проверяют истинность условия отсутствия стены у каждой стороны той клетки, где находится Робот:

сверху свободно

снизу свободно

слева свободно

справа свободно

Эти команды можно использовать вместе с условием «если», имеющим следующий вид:

если <условие> то

последовательность команд

все

Последовательность команд – это одна или несколько любых команд, выполняемых Роботом. Например, для передвижения на одну клетку вправо, если справа нет стенки, можно использовать такой алгоритм:

если справа свободно то

вправо

все

В одном условии можно использовать несколько команд, применяя логические связи **и**, **или**, **не**, например:

если (справа свободно) и (не снизу свободно) то

вправо

все

Для повторения последовательности команд можно использовать цикл «пока», имеющий следующий вид:

нц пока <условие>

последовательность команд

кц

Например, для движения вправо, пока это возможно, можно использовать следующий алгоритм:

нц пока справа свободно

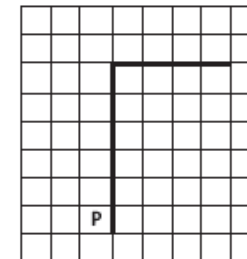
вправо
кц

Также у Робота есть команда **закрасить**, закрашивающая клетку, в которой Робот находится в настоящий момент.

Выполните задание.

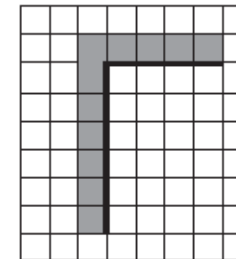
На бесконечном поле имеется вертикальная стена. **Длина стены неизвестна**. От верхнего конца стены вправо отходит горизонтальная стена **также неизвестной длины**. Робот находится в клетке, расположенной слева от нижнего края вертикальной стены.

На рисунке указан один из возможных способов расположения стен и Робота (Робот обозначен буквой «Р»).



Напишите для Робота алгоритм, закрашивающий все клетки, расположенные левее вертикальной стены и выше горизонтальной стены и прилегающие к ним. Робот должен закрасить только клетки, удовлетворяющие данному условию.

Например, для приведённого выше рисунка Робот должен закрасить следующие клетки (см. рисунок).



Конечное расположение Робота может быть произвольным. Алгоритм должен решать задачу для произвольного размера поля и любого допустимого расположения стен внутри прямоугольного поля. При исполнении алгоритма Робот не должен разрушиться.

Алгоритм напишите в текстовом редакторе и сохраните в текстовом файле.

Название файла и каталог для сохранения Вам сообщит учитель.

- 9.2** Напишите программу, которая в последовательности целых чисел определяет количество чётных чисел, кратных 7. Программа получает на вход целые числа, количество введённых чисел неизвестно, последовательность чисел заканчивается числом 0 (0 – признак окончания ввода, не входит в последовательность).
Количество чисел не превышает 1000. Введённые числа по модулю не превышают 30 000. Программа должна вывести одно число: количество чётных чисел, кратных 7.

Пример работы программы:

Входные данные	Выходные данные
–32 14 17 0	1

Ответы к заданиям

№ задания	Ответ
1	4
2	ЕЖИ
3	2
4	9
5	56
6	8
7	12121
8	ВСХВУМ

Критерии оценивания заданий с развёрнутым ответом

9.1

Исполнитель Робот умеет перемещаться по лабиринту, начерченному на плоскости, разбитой на клетки. Ниже приведено описание Робота.

У Робота есть четыре команды перемещения:

вверх

вниз

влево

вправо

При выполнении любой из этих команд Робот перемещается на одну клетку соответственно: вверх ↑, вниз ↓, влево ←, вправо →.

Между соседними (по сторонам) клетками может стоять стена, через которую Робот пройти не может. Если Робот получает команду передвижения через стену, то он разрушается.

Четыре команды проверяют истинность условия отсутствия стены у каждой стороны той клетки, где находится Робот:

сверху свободно

снизу свободно

слева свободно

справа свободно

Эти команды можно использовать вместе с условием «**если**», имеющим следующий вид:

если <условие> **то**

последовательность команд

все

«Последовательность команд» – это одна или несколько любых команд, выполняемых Роботом. Например, для передвижения на одну клетку вправо, если справа нет стенки, можно использовать такой алгоритм:

если *справа свободно* **то**

вправо

все

В одном условии можно использовать несколько команд, применяя логические связки **и**, **или**, **не**, например:

если (*справа свободно*) **и** (*не снизу свободно*) **то**

вправо

все

Для повторения последовательности команд можно использовать цикл «**пока**», имеющий следующий вид:

нц *пока* <условие>

последовательность команд

кц

Например, для движения вправо, пока это возможно, можно использовать следующий алгоритм:

нц *пока* *справа свободно*

вправо

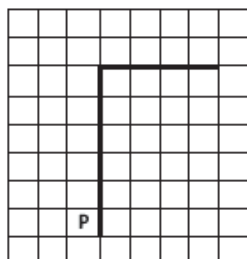
кц

Также у Робота есть команда **закрасить**, закрашивающая клетку, в которой Робот находится в настоящий момент.

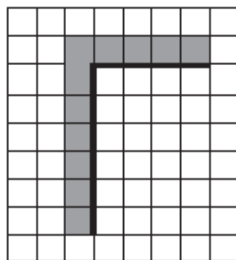
Выполните задание.

На бесконечном поле имеется вертикальная стена. **Длина стены неизвестна**. От верхнего конца стены вправо отходит горизонтальная стена **также неизвестной длины**. Робот находится в клетке, расположенной слева от нижнего края вертикальной стены.

На рисунке указан один из возможных способов расположения стен и Робота (Робот обозначен буквой «Р»):



Напишите для Робота алгоритм, закрашивающий все клетки, расположенные левее вертикальной стены и выше горизонтальной стены и прилегающие к ним. Робот должен закрасить только клетки, удовлетворяющие данному условию. Например, для приведённого выше рисунка Робот должен закрасить следующие клетки (см. рисунок):



Конечное расположение Робота может быть произвольным. Алгоритм должен решать задачу для произвольного размера поля и любого допустимого расположения стен внутри прямоугольного поля. При исполнении алгоритма Робот не должен разрушиться. Алгоритм напишите в текстовом редакторе и сохраните в текстовом файле. Название файла и каталог для сохранения вам сообщат организаторы экзамена.

Содержание верного ответа и указания по оцениванию (допускаются иные формулировки ответа, не искажающие его смысла)	
Команды исполнителя будем записывать жирным шрифтом, а комментарии, поясняющие алгоритм и не являющиеся его частью, – курсивом. Начало комментария будем обозначать символом « ». <i>Двигаемся вверх, пока не дойдём до конца вертикальной стены, закрашивая все клетки на пути.</i> нц пока не справа свободно закрасить вверх кц <i>Закрасим угловую клетку и переместимся в начало горизонтальной стены.</i> закрасить вправо <i>Двигаемся вправо до конца горизонтальной стены, закрашивая все клетки на пути.</i> нц пока не снизу свободно закрасить вправо кц Возможны и другие варианты решения	
Указания по оцениванию	Баллы
Записан правильный алгоритм, не приводящий к уничтожению Робота, полностью решающий поставленную задачу. Допускается использование иного синтаксиса инструкций исполнителя, более привычного учащимся	2
Алгоритм в целом записан верно, но может содержать одну ошибку. Примеры ошибок: 1) Робот закрашивает одну или несколько лишних клеток; 2) Робот не закрашивает одну из клеток (например, клетку над пересечением вертикальной и горизонтальной стен)	1
Задание выполнено неверно, или возможных ошибок в алгоритме больше одной	0
Максимальный балл	2

- 9.2** Напишите программу, которая в последовательности целых чисел определяет количество чётных чисел, кратных 7. Программа получает на вход целые числа, количество введённых чисел неизвестно, последовательность чисел заканчивается числом 0 (0 – признак окончания ввода, не входит в последовательность).
Количество чисел не превышает 1000. Введённые числа по модулю не превышают 30 000.
Программа должна вывести одно число: количество чётных чисел, кратных 7.

Пример работы программы:

Входные данные	Выходные данные
-32 14 17 0	1

Содержание верного ответа и указания по оцениванию (допускаются иные формулировки ответа, не искажающие его смысла)		
Решением является программа, записанная на любом языке программирования. Пример верного решения, записанного на языке Паскаль: <pre>var a, answer: integer; begin answer:=0; readln(a); while a<>0 do begin if (a mod 2 = 0) and (a mod 7 = 0) then answer := answer + 1; readln(a); end; writeln(answer) end.</pre> Возможны и другие варианты решения. Для проверки правильности работы программы необходимо использовать следующие тесты:		
№	Входные данные	Выходные данные
1	10 7 0	0
2	17 0	0
3	14 0	1
4	-42 1001 2800 0	2

Указания по оцениванию	Баллы
Предложено верное решение. Программа правильно работает на всех приведённых выше тестах. Программа может быть записана на любом языке программирования	2
Программа выдаёт неверный ответ на одном из тестов, приведённых выше. Например, решение, в котором неверно задано условие отбора чисел: (a mod 2=0) or (a mod 7=0), выдаст неправильный ответ на тесте № 1. ИЛИ Программа выдаёт на всех тестах ответ на единицу больше, чем требуется. Такое возможно, если в решении при определении количества чётных чисел, кратных 7, ошибочно учитывается 0	1
Программа выдаёт на тестах неверные ответы, отличные от описанных в критерии на 1 балл	0
Максимальный балл	2