

«Мир информатики»

Журнал для тех, у кого информатика – любимый школьный предмет

Выпуск № 48, сентябрь 2020 г.

011000100010101001001111100100011001001000101001001001011

ЕГЭ

Методика выполнения задания 5 демонстрационного варианта ЕГЭ по информатике и ИКТ 2021 года

В Спецификации контрольных измерительных материалов для проведения в 2021 году единого государственного экзамена по информатике и ИКТ применительно к заданию 5 в качестве проверяемого элемента содержания указано: «Формальное исполнение алгоритма, записанного на естественном языке, или умение создавать линейный алгоритм для формального исполнителя с ограниченным набором команд».

Обсудим методику выполнения указанного задания, а также аналогичных заданий из демонстрационных вариантов ЕГЭ нескольких последних лет.

Задание из [5]

Условие

Автомат получает на вход трехзначное число. По этому числу строится новое число по следующим правилам.

1. Складываются первая и вторая, а также вторая и третья цифры исходного числа.
2. Полученные два числа записываются друг за другом в порядке убывания (без разделителей).

Пример. Исходное число: 348. Суммы: $3 + 4 = 7$; $4 + 8 = 12$. Результат: 127.

Укажите наименьшее число, в результате обработки которого автомат выдаст число 1711.

Решение

Так как результат работы алгоритма – четырехзначный, то одна из сумм равна 17, а вторая – 11. Число 17 может быть суммой только двух цифр – 8 и 9. Одна из этих цифр «участвует» во второй сумме (11), то есть является «средней» в записи входного числа. Это значит, что возможен один из двух вариантов входного числа:

- 1) ?8?
- 2) ?9?

Поскольку вторая сумма цифр равна 11, то неизвестная третья цифра входного числа может быть равна, соответственно, 3 или 2.

По условию следует найти минимальное значение входного числа, то именно одна из этих цифр является в нем первой. Следовательно, возможны два варианта искомого числа:

- 1) 389;
- 2) 298,

из которых минимальное значение – второе.

Итак, ответ: 298.

Задание из [4]

Условие

На вход алгоритма подаётся натуральное число N . Алгоритм строит по нему новое число R следующим образом:

- 1) строится двоичная запись числа N ;
- 2) К этой записи дописываются справа ещё два разряда по следующему правилу:
 - а) складываются все цифры двоичной записи числа N , и остаток от деления суммы на 2 дописывается в конец числа (справа). Например, запись 11100 преобразуется в запись 111001;
 - б) над этой записью производятся те же действия – справа дописывается остаток от деления суммы её цифр на 2.

Полученная таким образом запись (в ней на два разряда больше, чем в записи исходного числа N) является двоичной записью искомого числа R .

Укажите минимальное число R , которое превышает число 83 и может являться результатом работы данного алгоритма. В ответе это число запишите в десятичной системе счисления.

Решение

Проанализируем условие. В нем фигурируют два десятичных числа:

- 1) исходное число N ;
- 2) результат работы алгоритма R .
- 3) некоторое целое число (в данном случае – 83), которое назовем «граничным» и обозначим G .

Если соответствующие двоичные записи перечисленных чисел обозначим N_2 , R_2 и G_2 , то решение задачи сводится к двум этапам:

- 1) нахождение такого минимального значения N_2 , которое в результате выполнения алгоритма даст значение R_2 , большее G_2 ;
- 2) определение десятичного эквивалента найденного значения N_2 .

Значение G_2 получить легко. Пусть оно имеет вид (условный):

1	0	0	1	...			
---	---	---	---	-----	--	--	--

k цифр

Про значение R_2 мы знаем, что оно имеет вид:

--	--	--	--	--	--	--	--

где оттененная часть совпадает со значением N_2 .

Сделаем важное для решения утверждение. Длина (количество цифр) значения R_2 , соответствующего искомому R , должна быть не менее длины числа G_2 . Это значит, что поиск значения N_2 можно начинать с числа, состоящего из $(k - 2)$ цифр значения G_2 .

Переведем 83 в двоичную систему – получим значение 1010011.

Согласно утверждению, сделанному чуть выше, минимально возможное значение N_2 равно 10011. Исследуем это значение:

N_2	Кол-во единиц	Двоичная запись результата (R_2)	Значение R
10100	Четное	1010000	80
10101	Нечетное	1010110	86

Так как $86 > 83$, то решение найдено.

Следовательно, ответ: 86.

Итак, алгоритм решения обсуждаемой задачи можно сформулировать следующим образом:

1) по заданному граничному числу G получить $G2$;

2) отбросить в $G2$ две последние цифры;

3) принимая новое и последующие за ним двоичные числа в качестве числа $N2$, приписывать к каждому две новые цифры по правилу:

если количество единиц в числе $N2$ нечетное,

то

к нему приписываются цифры 1 и 0

иначе

к нему приписываются цифры 0 и 0

все

Первое число $N2$, дающее результат $R2 > G2$, определяет искомое значение R .

Задания для самостоятельной работы

1. Для условия, приведенного в начале раздела, определите такое наименьшее число R , для которого может являться результатом работы данного алгоритма и превышает:

а) 25;

б) 41;

в) 88.

2. На вход алгоритма подаётся натуральное число N . Алгоритм строит по нему новое число R следующим образом:

1) строится двоичная запись числа N ;

2) К этой записи дописываются справа ещё два разряда по следующему правилу:

а) если остаток от деления суммы всех цифр двоичной записи числа N на 2 равен нулю, то в конец в конец числа (справа) дописывается 1, в противном случае – дописывается 0. Например, запись 11000 преобразуется в запись 110001;

б) над новой записью производятся те же действия.

Полученная таким образом запись (в ней на два разряда больше, чем в записи исходного числа N) является двоичной записью числа R .

Укажите минимальное число R , которое может являться результатом работы данного алгоритма и превышает:

а) 25;

б) 41;

в) 88.

Задание из [3]

Условие

На вход алгоритма подаётся натуральное число N . Алгоритм строит по нему новое число R следующим образом:

1) строится двоичная запись числа N ;

2) к этой записи дописываются справа ещё два разряда по следующему правилу: если N чётное, в конец числа (справа) дописывается сначала ноль, а затем единица. В противном случае, если N нечётное, справа дописывается сначала единица, а затем ноль.

Например, двоичная запись 100 числа 4 будет преобразована в 10001, а двоичная запись 111 числа 7 будет преобразована в 11110. Полученная таким образом запись (в ней на два разряда больше, чем в записи исходного числа N) является двоичной записью числа R – результата работы данного алгоритма.

Укажите минимальное число R , которое больше 102 и может являться результатом работы данного алгоритма. В ответе это число запишите в десятичной системе счисления.

Решение

Анализ показывает, что указанная задача отличается от рассмотренной в предыдущем разделе чуть измененным алгоритмом преобразования заданного числа и решать ее можно аналогично.

Прежде чем проводить анализ, заметим, что у четного числа последняя двоичная цифра – 0, у нечетного – 1.

Переведем 102 в двоичную систему – получим значение 1100110

Таблица для анализа:

$N2$	Число N	Двоичная запись результата ($R2$)	Значение R
11001	Нечетное	1100110	102
11010	Четное	1101001	105

Следовательно, ответ: 105.

Задание из [4]

Условие

На вход алгоритма подаётся натуральное число N . Алгоритм строит по нему новое число R следующим образом:

- 1) строится двоичная запись числа N ;
- 2) К этой записи дописываются справа ещё два разряда по следующему правилу:
 - а) складываются все цифры двоичной записи числа N , и остаток от деления суммы на 2 дописывается в конец числа (справа). Например, запись 11100 преобразуется в запись 111001;
 - б) над этой записью производятся те же действия – справа дописывается остаток от деления суммы её цифр на 2.

Полученная таким образом запись (в ней на два разряда больше, чем в записи исходного числа N) является двоичной записью искомого числа R .

Укажите минимальное число R , которое превышает число 97 и может являться результатом работы данного алгоритма. В ответе это число запишите в десятичной системе счисления.

Решение

Оно аналогично решению задачи в 2018

Переведем 97 в двоичную систему – получим значение 1100001.

Таблица для анализа:

$N2$	Кол-во единиц	Двоичная запись результата ($R2$)	Значение R
11000	Четное	1100000	80
11001	Нечетное	1100110	102

Итак, ответ: 102.

Задание из [5]

Условие

На вход алгоритма подается натуральное число N . Алгоритм строит по нему новое число R следующим образом.

1. Строится двоичная запись числа N .
2. К этой записи дописываются справа еще два разряда по следующему правилу:
 - а) складываются все цифры двоичной записи числа N , и остаток от деления суммы на 2 дописывается в конец числа (справа). Например, запись 11100 преобразуется в запись 111001;
 - б) над этой записью производятся те же действия – справа дописывается остаток от деления суммы ее цифр на 2.

Полученная таким образом запись (в ней на два разряда больше, чем в записи исходного числа N) является двоичной записью искомого числа R .

Укажите такое наименьшее число N , для которого результат работы данного алгоритма больше числа 77. В ответе это число запишите в десятичной системе счисления.

Решение

Обратим внимание на то, что данная задача отличается от трех предыдущих тем, что следует найти не число, являющееся результатом работы алгоритма, а «входное» число алгоритма.

Проанализируем условие. В нем фигурируют три десятичных числа:

- 1) исходное число N ;
- 2) результат работы алгоритма R ;
- 3) некоторое целое число (в данном случае – 77), которое назовем «граничным» и обозначим G .

Согласно условию, нужно найти такое минимальное значение N , которое в результате выполнения алгоритма даст значение R , большее G .

Если соответствующие двоичные записи перечисленных чисел обозначим N_2 , R_2 и G_2 , то решение задачи сводится к двум этапам:

- 1) нахождение такого минимального значения N_2 , которое в результате выполнения алгоритма даст значение R_2 , большее G_2 ;
- 2) определение десятичного эквивалента найденного значения N_2 .

Обсудим первый этап. Здесь также можно использовать утверждение, сделанное в 2018 – длина (количество цифр) значения R_2 , соответствующего искомому N_2 , должна быть не менее длины числа G_2 . Это значит, что поиск значения N_2 можно начинать с числа, состоящего из $(k - 2)$ цифр значения G_2 .

Начнем решение.

Переведем число 77 в двоичную систему – 1001101. Согласно утверждению, сделанному чуть выше, минимально возможное значение N_2 равно 10011. Исследуем это значение:

N_2	Кол-во единиц	Двоичная запись результата (R_2)	Значение R ¹
10011	Нечетное	1001110	78

Итак, значение 10011 дает нужный результат, то есть ответ: $N = 19$.

Обратим внимание на то, что в рассмотренном случае найденное число N_2 имеет вид 10011, что совпадает с укороченной на две цифры двоичной записью заданного граничного числа 77. Однако это не означает, что значение N_2 (а следовательно, и N) можно получить, отбросив два последние цифры двоичной записи заданного в условии числа G ! Подтвердим сказанное на примере.

Пусть нужно определить такое наименьшее число N , для которого результат работы описанного в условии алгоритма больше числа 90.

$$90_{10} = 1011010_2.$$

Отбросив последние цифры, получим 10110. При этом, согласно алгоритму, результат R будет равен 90 ($1011010_2 = 90_{10}$), который не подходит по условию.

Проверка следующего числа дает искомый результат:

N_2	Кол-во единиц	R_2	R
10111	Четное	1011100	92

– то есть правильный ответ – 23 (10111_2).

¹ Это значение можно не определять, а сравнивать R_2 и G_2 .

Итак, алгоритм решения обсуждаемой задачи можно сформулировать следующим образом:

- 1) по заданному граничному числу G получить $G2$;
- 2) отбросить в $G2$ две последние цифры;
- 3) принимая новое и последующие за ним двоичные числа в качестве числа $N2$, приписывать к каждому две новые цифры по правилу:

если количество единиц в числе $N2$ нечетное,

то

к нему приписываются цифры 1 и 0

иначе

к нему приписываются цифры 0 и 0

все

Первое число $N2$, дающее результат $R2 > G2$, определяет искомое значение N^2 .

Задания для самостоятельной работы

1. Для условия, приведенного в начале раздела, определите такое наименьшее число N , для которого результат работы данного алгоритма больше числа:

- а) 25;
- б) 41;
- в) 88.

2. На вход алгоритма подаётся натуральное число N . Алгоритм строит по нему новое число R следующим образом:

1) строится двоичная запись числа N ;

2) К этой записи дописываются справа ещё два разряда по следующему правилу:

а) если остаток от деления суммы всех цифр двоичной записи числа N на 2 равен нулю, то в конец в конец числа (справа) дописывается 1, в противном случае – дописывается 0. Например, запись 11000 преобразуется в запись 110001;

б) над новой записью производятся те же действия.

Полученная таким образом запись (в ней на два разряда больше, чем в записи исходного числа N) является двоичной записью искомого числа R .

Укажите такое наименьшее число N , для которого результат работы данного алгоритма больше числа:

- а) 21;
- б) 56;
- в) 83.

Методика выполнения задания 10 демонстрационного варианта ЕГЭ по информатике и ИКТ 2021 года

В Спецификации контрольных измерительных материалов для проведения в 2021 году единого государственного экзамена по информатике и ИКТ применительно к заданию 5 в качестве проверяемого элемента содержания указано: «Информационный поиск средствами операционной системы или текстового процессора».

Условие

С помощью текстового редактора определите, сколько раз, не считая сносок, встречается слово «долг» или «Долг» в тексте романа в стихах А.С. Пушкина «Евгений Онегин». Другие формы слова «долг», такие как «долги», «долгами» и т. д., учитывать не следует. В ответе укажите только число.

² Можно доказать, что для нахождения искомого значения требуется проверка не более двух чисел.

Указанная задача может быть достаточно быстро решена путём использования такой возможности текстового редактора Microsoft Word, как поиск в документе нужных фрагментов текста (слов или их части). Запрос на поиск может быть достаточно сложным, включающим учет регистра букв (строчные или прописные), длину слов, комбинацию символов и т. д.

Обсудим основные, простые, возможности.

Для поиска с простым запросом достаточно вызвать панель Навигация. Для этого в ленте Word надо переключиться на вкладку Главная и нажать кнопку Заменить в правой части вкладки (см. рис. 1).

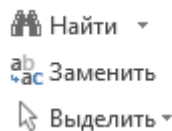


Рис. 1

Можно также нажать комбинацию клавиш **Ctrl** и **F**.

В результате в левой части экрана появится панель Навигация:

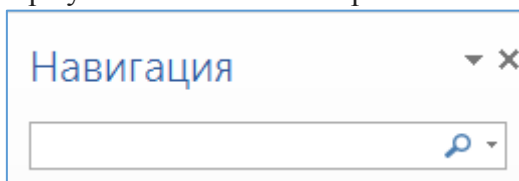


Рис. 2

То, что надо найти, указывается в поле, в котором изображена лупа. Можно также выделить нужное слово или его часть, после чего вызвать панель Навигация.

Итак, начнем.

Для поиска всех вхождений, например, последовательности букв *порт*, следует записать эту последовательности в окне поиска³ и нажать клавишу **Enter**. Будут найдены и выделены цветом все имеющиеся в документе слова *порт*, а также другие слова, в которых имеется указанная последовательности букв (*спорт, портал, паспорт, портативный* и т. п.). Их общее количество (нужное в нашем случае значение) будет указано в верхней части панели Навигация (рис. 3).

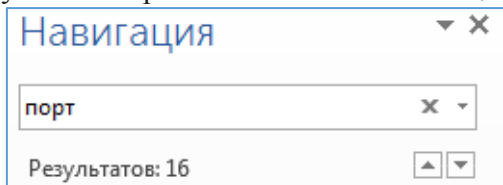


Рис. 3

Обратим внимание на то, что слова, в которых буква п – прописная (Порт, Портал и т. п.), учтены не будут. Чтобы учесть только их, следует в поле поиска записать Порт.

Обсудим, что будет найдено, если записать в поле поиска порт?. Как, очевидно, известно читателю, в масках (шаблонах) для поиска файлов по их именам символ «?» означает – «один любой символ». Такой же смысл он имеет и при поиске, то есть будут найдены и выделены, например, такие фрагменты: порталы, спортивный, портг.⁴ и т. п.

А если записать в поле поиска порт*? Применительно к именам файлов символ «звездочка» означает «любая допустимая последовательность символов». При поиске этот символ также имеет такой же смысл, только в результате поиска будут выделена только последовательности букв порт, то есть так же, как и при поиске без символа * (естественно, что и количество найденных вхождений будет тем же).

³ Или, как указывалось чуть выше, перед открытием панели Навигация предварительно выделить нужную последовательность букв в тексте.

⁴ Точка также будет выделена цветом.

Хотя выше уже отмечалось, как определить количество вхождений, начинающихся с прописной буквы, расскажем о другом способе, который понадобится нам для более сложных поисков. В этом случае возможны два варианта подготовительных действий:

1) после открытия панели Навигация путем нажатия клавиш **Ctrl** и **F** щелкнуть на треугольнике справа от изображения лупы и в появившемся списке выбрать Расширенный список (см. рис. 4).

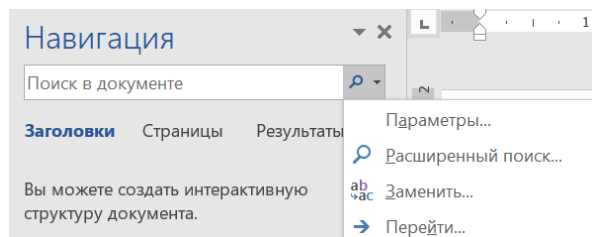


Рис. 4

Появится окно Найти и заменить с активной вкладкой Найти (см. рис. 5).

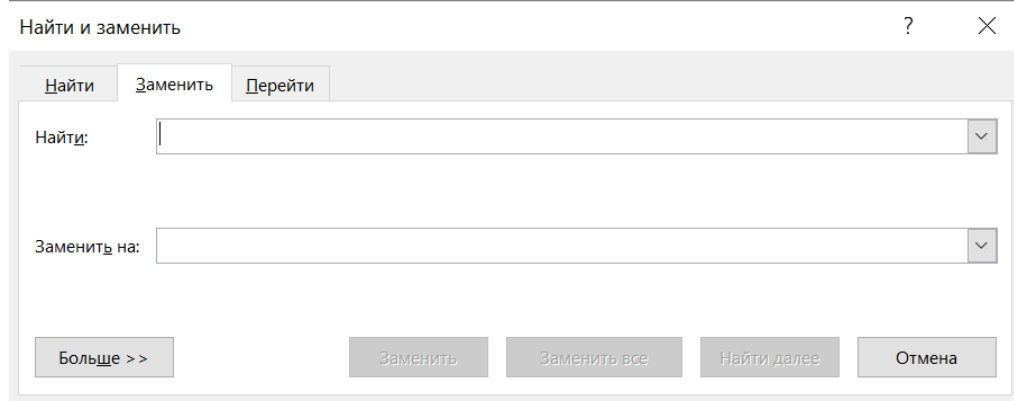


Рис. 5

2) в ленте Word переключиться на вкладку Главная и нажать кнопку Заменить в правой части вкладки – будет открыто окно Найти и заменить, в котором активной будет вкладка Заменить:

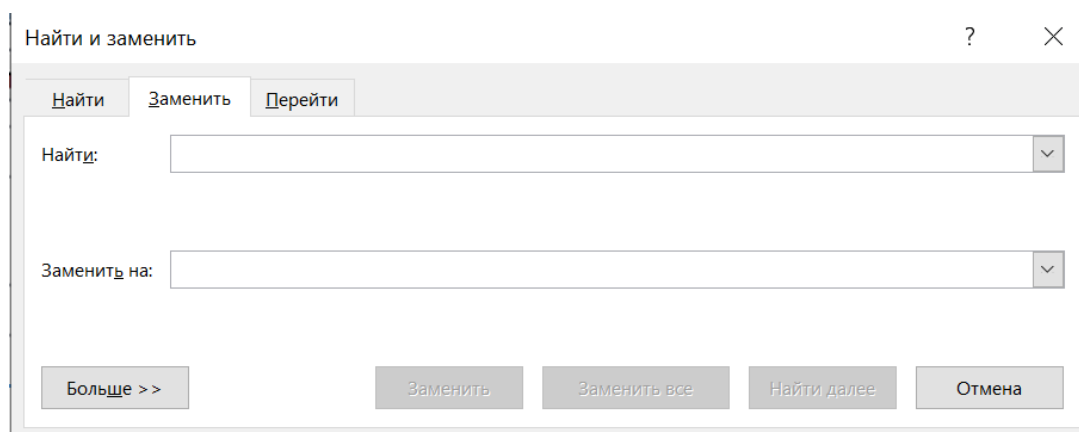


Рис. 6

Для поиска можно работать с активной вкладкой⁵, но лучше перейти на вкладку Найти.

В обоих вариантах подготовительных действий, чтобы при поиске учитывались тексты с прописными буквами (например, Порт, Портал, Портфель и т.п.), следует:

- 1) в поле ввода записать нужный текст;
- 2) поставить «галочку» в переключателе с надписью Учитывать регистр (см. рис. 7).

⁵ В этом случае подсчет искомых значений произойдет при одновременной их замены на «пустое» значение, то есть указанные значения будут удалены.

При этом активной станет кнопка Найти, по щелчку на которой (в пункте Основной документ) будут показаны все результаты поиска (рис. 7).

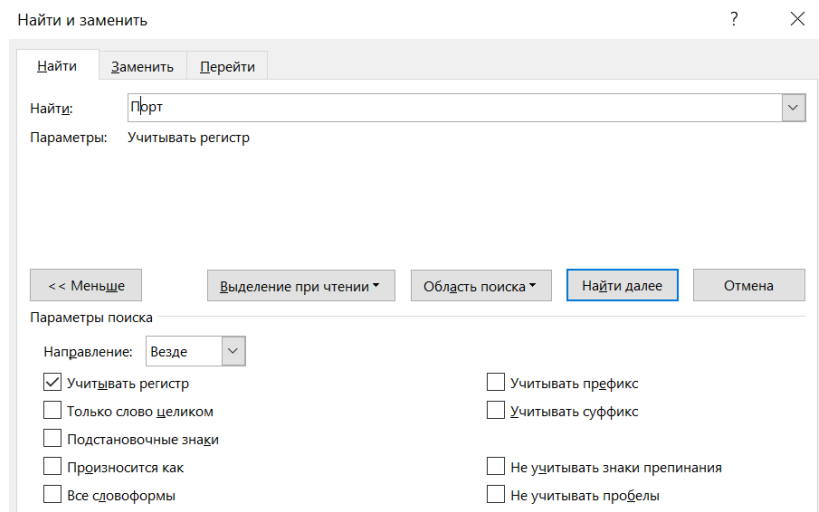


Рис. 7

Более сложный поиск проводится с использованием так называемых «регулярных выражений». Регулярное выражение — это запись, которая представляет собой условное обозначение критериев, которым должна соответствовать искомый текст. С помощью регулярных выражений можно найти множество слов или их частей, соответствующих заданным, при необходимости — достаточно сложным, условиям.

В данной статье опишем запросы на поиск с достаточно простыми условиями в регулярных выражениях.

Регулярное выражение для поиска записывается в поле Найти. В нем используются так называемые «подстановочные знаки». Их полный перечень можно увидеть, поставив в окне поиска «галочку» в переключателе с надписью Подстановочные знаки, после чего щелкнуть на кнопке Специальный в нижней части окна.

Можно также записывать знаки «вручную» (с помощью клавиатуры), но при использовании подстановочных знаков указанный переключатель должен быть обязательно включен.

Приведем несколько примеров.

1. Чтобы найти все слова, начинающиеся на «порт» (*портативный, портал, порт* и т. п., но не спорт), регулярное выражение должно быть таким:

<порт*

Внимание! При включенном переключателе с надписью Подстановочные знаки переключатель с надписью Учитывать регистр (рис. 7) будет недоступен, то есть слова, начинающиеся на «Порт» (*Портативный, Портал, Порт* и т. п.) учтены не будут.

Чтобы найти их (только!), регулярное выражение должно быть таким:

<Порт*

2. Регулярное выражение:

a>

– найдет все слова, оканчивающиеся на букву *a*.

Как, конечно, уже понял читатель, знак «<>» (меньше) означает – «начало слова», а знак «<>>» (больше) – «конец слова».

3. Чтобы найти все слова, начинающиеся на любую букву, за которой следует буква *a*, регулярное выражение имеет вид:

<?a

Задания для самостоятельной работы

Определите количество вхождений в документ текстового редактора Microsoft Word:

- 1) буквы *ю*;
- 2) последовательностей букв *поле, поли, полк, пола, полу, полы, поля*;
- 3) начинающихся на *без*;
- 4) оканчивающихся на *лан*;
- 5) начинающихся на две любые буквы, за которыми следует буква *о*;
- 6) слов *мир*;
- 7) слов *марка* и *Марка*.

Документы для выполнения заданий подготовьте самостоятельно.

Методика выполнения задания 12 демонстрационного варианта ЕГЭ по информатике и ИКТ 2021 года

В Спецификации контрольных измерительных материалов для проведения в 2021 году единого государственного экзамена по информатике и ИКТ применительно к заданию 5 в качестве проверяемого элемента содержания указано: «Умение анализировать результат исполнения алгоритма».

Условие

Исполнитель Редактор получает на вход строку цифр и преобразовывает 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 70 идущих подряд цифр 8? В ответе запишите полученную строку.

НАЧАЛО

ПОКА нашлось (2222) **ИЛИ** нашлось (8888)

ЕСЛИ нашлось (2222)

ТО заменить (2222, 88)

2. Дана программа для исполнителя **Редактор**:

НАЧАЛО

ПОКА нашлось (25) **ИЛИ** нашлось (355) **ИЛИ** нашлось (4555)

ЕСЛИ нашлось (25)

ТО заменить (25, 4)

КОНЕЦ ЕСЛИ

ЕСЛИ нашлось (355)

ТО заменить (355, 2)

КОНЕЦ ЕСЛИ

ЕСЛИ нашлось (4555)

ТО заменить (4555, 3)

КОНЕЦ ЕСЛИ

КОНЕЦ ПОКА

КОНЕЦ

Какая строка получится в результате применения приведенной выше программы к строке, состоящей из цифры 2 и следующих за ними 81 цифр 5?

Литература

1. Демонстрационный вариант контрольных измерительных материалов единого государственного экзамена 2021 года по информатике и ИКТ. / <http://fipi.ru/ege-i-gve-11/demoversii-specifikacii-kodifikatory> (Демонстрация, спецификации, кодификаторы 2021. Информатика и ИКТ).

2. Демонстрационный вариант контрольных измерительных материалов единого государственного экзамена 2020 года по информатике и ИКТ. / <http://fipi.ru/ege-i-gve-11/demoversii-specifikacii-kodifikatory> (Демонстрация, спецификации, кодификаторы 2020. Информатика и ИКТ).

3. Демонстрационный вариант контрольных измерительных материалов единого государственного экзамена 2019 года по информатике и ИКТ. / <http://fipi.ru/ege-i-gve-11/demoversii-specifikacii-kodifikatory> (Демонстрация, спецификации, кодификаторы 2019. Информатика и ИКТ).

4. Демонстрационный вариант контрольных измерительных материалов единого государственного экзамена 2018 года по информатике и ИКТ. / <http://fipi.ru/ege-i-gve-11/demoversii-specifikacii-kodifikatory> (Демонстрация, спецификации, кодификаторы 2018. Информатика и ИКТ).

5. Демонстрационный вариант контрольных измерительных материалов единого государственного экзамена 2017 года по информатике и ИКТ. / <http://fipi.ru/ege-i-gve-11/demoversii-specifikacii-kodifikatory> (Демонстрация, спецификации, кодификаторы 2017. Информатика и ИКТ).

Книжная полка

Книга для подготовки к ЕГЭ по информатике

Издательство ДМК Пресс предлагает книгу Д.М. Златопольского «Подготовка к ЕГЭ по информатике в 2020 году. Решение задач по программированию».



Книга предназначена для подготовки учащихся к Единому государственному экзамену по информатике и ИКТ в части решения задач по программированию. Рассмотрена методика решения основных типовых задач, а также заданий из демонстрационных вариантов ЕГЭ (в том числе варианта 2020 года) и из пособий, написанных разработчиками контрольно-измерительных материалов по информатике.

Издание будет полезно также студентам вузов и колледжей, преподавателям информатики и другим читателям при изучении программирования вне связи с ЕГЭ.

С оглавлением книги и содержанием первой главы можно ознакомиться по адресу:

<https://dmkpress.com/files/PDF/978-5-97060-829-6.pdf>

Сборник задач по программированию

Издательство ДМК Пресс предлагает книгу Д.М. Златопольского «1400 задач по программированию».



В книге приведено более 1400 задач по программированию. Задачи имеют разный уровень сложности и охватывают большой круг тем. Они не привязаны к конкретному языку программирования и могут быть решены в программе на любом языке (Паскаль, Python, Си, школьном алгоритмическом языке и т. д.).

Издание будет полезно как читателям, изучающим программирование самостоятельно, так и преподавателям школ, колледжей и вузов. В последнем случае задачи могут быть использованы при объяснении нового материала, для организации самостоятельной работы учащихся и проведения контрольных мероприятий. Ряд задач предназначен для внутришкольных олимпиад, для использования в кружках по программированию и т. п.

По издательской цене (гораздо дешевле, чем в магазинах, даже с учетом стоимости доставки) обе книги можно приобрести через сайт издательства <https://dmkpress.com/>

Задачник

Три очень умных попугая

На рыночной площади одного маленького городка стоял старика с тремя очень умными говорящими попугаями. За некоторую плату каждый попугай отвечал на заданные ему вопросы. Жители городка знали, что один из попугаев при любой плате говорит правду, другой, сколько ему не дай, соврет, а третий отвечает согласно плате и настроению: решит, что мало дали, – соврет, решит, что плата достаточная, – скажет правду. Но вот кто из них – лжец, кто хитрец никто не знал, пока в город не приехал... специалист по логике.

- Кто стоит рядом с тобой? – спросил он у первого попугая.
- Лжец, – ответил тот.
- А ты кто такой? – спросил специалист у второго попугая.
- Я хитрец!
- А кто стоит рядом с тобой? – спросил он у третьего попугая.
- Абсолютно честный попугай! – был ответ.

После этого специалист по логике сказал, кто из них действительно абсолютно честный, кто лжец, а кто хитрец. А вы сможете это сделать?

Петя и Митя

На ступенях дома сидят рядышком два мальчика: Петя и Митя.

- Меня зовут Митя, – сказал мальчик с черными волосами.
- Меня зовут Петя, – сказал мальчик с белыми волосами.

Можно ли определить, как зовут каждого из мальчиков, если по крайней мере один из них говорит неправду?

Две бочки

Два бочки, по 10 галлонов каждая, снабжены этикетками А и Б. Бочка А содержала больше воды, чем бочка Б.

Сначала из бочки А в бочку Б перелили столько воды, сколько там уже было. После этого перелили из бочки Б в бочку А перелили столько жидкости, сколько в последней осталось. Наконец, из бочки А в бочку Б перелили столько, сколько теперь осталось в бочке Б. После этого в обеих бочках стало по 48 пинт (в галлоне чуть меньше 10 пинт) воды. А сколько ее было в каждой из бочек вначале?

Расхититель казны. Задача 2

Один король заподозрил своих министров в расхищении казны. Решив, что тот, у кого денег больше чем в казне, — виновен, он пригласил детектива для расследования. Однако каждый министр дал детективу взятку и попросил не называть сумму его состояния. Детектив взятки взял, но королю солгать не посмел. Он сказал так: “Казна больше на половину, чем все, что есть у первого и второго министров. Если же сложить имущество первого, второго и третьего министров, то эта сумма будет на три четверти больше казны”. Исходя из этого сообщения, король решил, что:

- 1) расхититель — второй министр;
- 2) расхититель — третий министр;
- 3) расхититель — первый министр;
- 4) определенно установить, кто расхититель, нельзя.

Какое решение короля верное?

Шестеро друзей

Шестеро друзей в ожидании электрички заскочили в буфет.

Маша взяла то же, что и Егор, и вдобавок ещё бутерброд с сыром.

Аня купила, то же, что и Саша, но не стала покупать шоколадное печенье.

Кирилл ел то же, что и Мила, но без луковых чипсов.

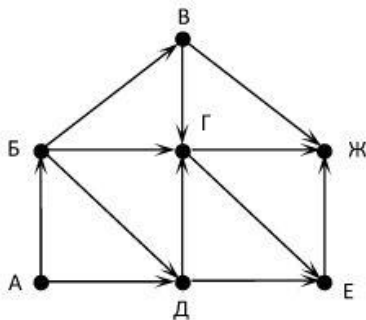
Егор завтракал тем же что и Аня, но бутерброду с котлетой предпочел картофельные чипсы.

Саша ел то же, что и Мила, но вместо молочного коктейля пил лимонад.

Из чего состоял завтрак каждого из друзей?

Из города А в город Ж

На рисунке ниже приведена схема дорог, связывающих города А, Б, В, Г, Д, Е, Ж, З, И, К. По каждой дороге можно двигаться только в одном направлении, указанном стрелкой. Сколько существует различных путей из города А в город Ж?



Этот зловредный “bug”

**В.В. Шилов,
Москва**

Легенды о происхождении тех или иных компьютерных терминов многочисленны и крайне живучи. Но все рекорды бьет одна из них – о слове *баг* (*bug*), под которым понимают любую ошибку в аппаратуре или программе (отсюда же *debugging* – отладка). В десятках журнальных и газетных статей, книг, словарей и энциклопедий можно прочесть нечто вроде: “Однажды в середине 1940-х гг., в работе предка современных компьютеров, релейной вычислительной машины Mark I, которую строили в Гарвардском университете, произошел сбой. Его причиной стал мотылек, который забился в одно из реле. Дежурный инженер извлек мотылька пинцетом, и с тех пор гарвардские ученые, когда в машине возникали те или иные неисправности, говорили: «Давай поищем жучка (*bug*)». Постепенно этот термин прижился и получил широкое распространение”.

Особый вклад в популяризацию этой истории внесла выдающийся программист, первая женщина, дослужившаяся до звания адмирала флота США, Грейс М. Хоппер (1906–1992). Математик с университетским образованием, она преподавала в одном из колледжей Нью-Йорка, а во время Второй мировой войны добровольцем пошла в армию. Конечно, ее служба в армии (а точнее, на флоте) была связана со специальностью – математики занимались вычислением различных таблиц, необходимых морякам, летчикам и артиллеристам. Одна из первых построенных в США вычислительных машин – еще не электронных, а электромеханических (релейных) – Mark I как раз предназначалась для решения таких задач. Так лейтенант Грейс Хоппер, которая решала на этой машине свои уравнения, впервые вошла в мир вычислительной техники. Вошла, чтобы остаться в нем на всю жизнь.

По словам Грейс Хоппер, жизнь злосчастного мотылька оборвалась летом 1945 года, и это событие сослужило инженерам и математикам неплохую службу. Во-первых, пока мотылька вытаскивали из реле и пока реле почистили, они смогли немного передохнуть. А во-вторых, в их лексиконе появилось новое выражение. Когда в машинный зал входил грозный начальник – создатель Mark I Говард Айкен (1900–1973) – и спрашивал, почему его подчиненные не работают, а бьют баклуши, те дружно отвечали: “А мы очищаем машину от насекомых!” (по-английски “We are debugging the computer!”). Вот так, по словам Грейс Хоппер, и родилось слово *bug*.

Здесь надо заметить, что описанное выше событие действительно имело место и даже было задокументировано! Тот самый мотылек был не только аккуратно вытасчен, но и высушен, и вклеен в рабочий журнал (сегодня этот журнал находится в одном из компьютерных музеев – см. *рис. 1*). Пристальный взгляд на страницу журнала убеждает нас в том, что хорошо известно историкам: любое свидетельство очевидца не может считаться достоверным фактом, пока не подкреплено документами... Ведь из записи следует, что случилась вся эта история не летом 1945 года, а двумя годами позже – 9 сентября 1947 года.

Правда, не слишком понятно, почему мотылек (по-английски *moth*) был вдруг назван “жуком”! Ведь *bug* – это даже не просто насекомое, а насекомое кровососущее, в первую очередь клоп. Разумеется, инженер не обязан быть специалистом в энтомологии⁶, но разницу между бабочкой и клопом он все-таки обычно знает...

⁶ Энтомология – раздел зоологии, изучающий насекомых. – Прим. ред.

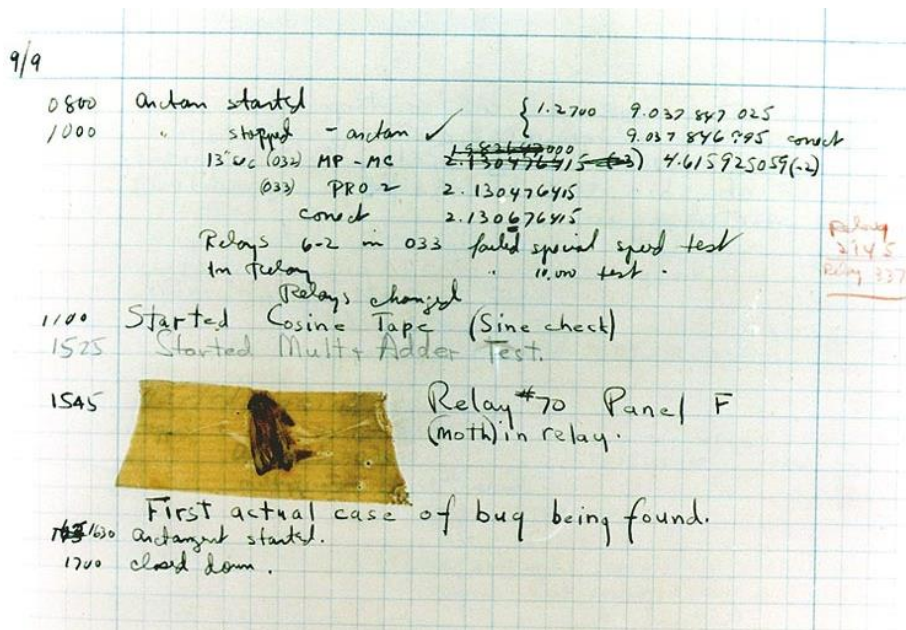


Рис. 1. Первый “bug” (страница из рабочего журнала)

Чтобы разобраться, давайте прочитаем соответствующую запись в рабочем журнале полностью: “1545 Реле #70 Приборная панель F (мотылек) в реле. Первый случай, когда был найден настоящий *bug*”. Последняя фраза неопровержимо свидетельствует о том, что слово *bug* к моменту обнаружения злосчастного мотылька уже было в ходу среди инженеров. Более того, обращение к различным источникам — техническим публикациям и словарям, показывает, что оно широко использовалось ими уже многие десятилетия!

И одним из первых его стал применять великий американский изобретатель Томас А. Эдисон (1847–1931). Он называл так различные неисправности, вызывающие ошибки в работе приборов и устройств – как небольшие, которые просто требуется устранить, так и более серьезные, причины которых еще только предстоит установить. Вполне допустимо предположить, что многочисленные и неистребимые мелкие неисправности вызывали у инженеров ассоциации со столь же надоедливым и неприятным насекомым – клопом! Так или иначе, известно, что Эдисон использовал слово *bug* в одном из своих писем, написанных еще в 1878 г. В 1889 г. оно впервые попало на страницы печати: репортер английской газеты “Pall Mall Gazette” 11 марта сообщил читателям, что “Мистер Эдисон провел две бессонных ночи, отыскивая *bug* в своем фонографе⁷” (на рис. 2 мы как раз видим расстроенного и задумавшегося изобретателя рядом с неисправным фонографом). Приблизительно в это же время слово *bug* было включено в словарь английского языка в значении “неисправность при работе телеграфной или другой электрической аппаратуры”.

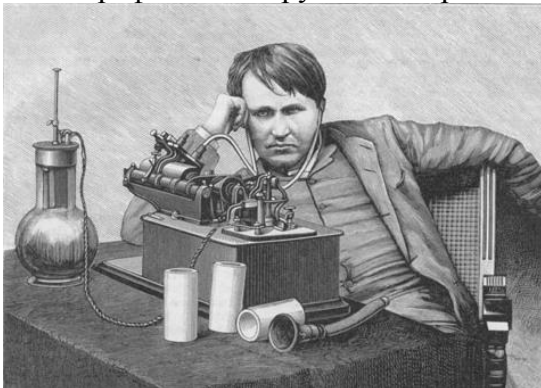


Рис. 2. Томас А. Эдисон (очевидно, озабоченный поиском неисправностей ☺)

⁷ Фонограф – первый прибор для записи и воспроизведения звука, изобретенный Эдисоном.

Таким образом, смысл приведенной выше записи в рабочем журнале становится вполне понятным. Инженеры, постоянно занятые устранением ошибок в работе оборудовании, называли их клопами (*bug*). И вдруг оказалось, что причиной ошибки в самом деле является насекомое (пусть даже совсем другого сорта)! Конечно, совпадение показалось забавным, что и было отмечено в рабочем журнале. И, скорее всего, если бы не Грейс Хоппер, то вряд ли этот эпизод привлек бы к себе чье-либо внимание.

Вообще говоря, живучесть легенд – факт крайне удивительный. Версию Грейс Хоппер не раз пытались опровергнуть, но безуспешно. Например, один из создателей электронно-вычислительной машины ENIAC Преспер Эккерт (1919–1995) в данном в 1986 г. интервью так ответил на вопрос, что он думает о происхождении термина *bug*: “Я знаю, что думает об этом Грейс Хоппер. Она много раз рассказывала свою сказку. Насколько я знаю, инженеры, – и механики, и электрики, называли так проблемы в работе оборудования задолго до того, как Грейс Хоппер услышала о самом существовании таких проблем. Это слово было новостью только для Грейс. Я никогда не говорил ей, что ее рассказ – это ерунда, но это все же полная ерунда. Слово давно было в ходу”.

Слова Эккерта подтверждает другая запись в рабочем журнале разработчиков Mark I. 17 апреля 1944 г. в нем были зафиксированы “проблемы с тестированием оборудования” и что Гарвард посетил консультант от компании IBM, чтобы “помочь нам обнаружить неисправности (*bugs*)”.

Да и сама Грейс Хоппер пользовалась этим словом задолго до гибели мотылька! На *рис. 3* показан лист с четырьмя шуточными рисунками, сделанными рукой самой Хоппер в июле 1945 года. На них изображены “баги” трех видов. Первый – “table worm” (т.е. “табличный червяк”), представленный в виде куска перфоленты, которая использовалась для ввода чисел в вычислительную машину. Ошибки на перфоленте, конечно, влекли за собой сбои в работе ЭВМ. Второй – это “Kitchie Boo Boo Bug”. Перевести это название можно, крайне приблизительно, как “дурацкая ошибка забодаю-забодаю”. Этот баг специализировался на выводе из строя реле. Третий, со строгим именем “NRL bug” (NRL – это Naval Research Laboratory, научно-исследовательская лаборатория военно-морского флота, в интересах которого велись вычисления), занимался тем, что портил результаты расчетов. А улыбающееся лицо в правом нижнем углу рисунка – это уже не баг, а “тот, кто выдает правильные данные”.

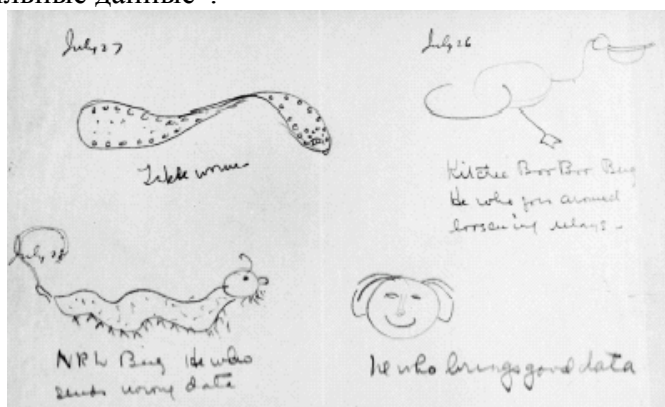


Рис. 3. Три вида багов ☺ (рисунок Грейс Хоппер)

Таким образом, вопреки распространенному мнению, вовсе не погибший мотылек дал имя компьютерному термину (или даже двум!). На самом деле это были давно известные и широко распространенные в среде инженеров слова. А о мотыльке никто не вспоминал несколько десятилетий – пока о нем не вспомнила адмирал Грейс Хоппер.

Впрочем, путь слова *bug* в компьютерный лексикон отнюдь не был устлан коврами дорожками. Потребовалось немало времени, чтобы слово из жаргона инженеров-электриков было окончательно “принято” всеми компьютерщиками. Так, в первом компьютерном словаре, составленном в 1951 г. Институтом радиоинженеров (IRE), оно отсутствовало. Вскоре в одном из журналов впервые появилось слово *debug*, которое объяснялось как

устранение неисправностей в работе аппаратуры компьютера. Спустя три года Г.Хоппер опубликовала свой словарь, в котором расширила значение понятия *debug*, включив в него также устранение ошибок в программах. Этот словарь лег в основу нескольких официальных изданий словаря компьютерных терминов, но слова *bug* ни в одном из них, как и у самой Хоппер, не было! И попало оно в него только в 1962 г.

Зато сегодня оно обрело права гражданства не только в английском, но и в других языках: например, во французском оно приняло вид *bogue*.



Рис. 4. Грейс Хоппер возле ЭВМ UNIVAC (начало 1950-х гг.)

Казалось бы, вопрос о происхождении популярного компьютерного термина полностью выяснен. Однако в действительности первыми не были не только Грейс Хоппер, но даже Эдисон и его коллеги. Известно, что слово *bug* в значении “непредусмотренное событие, имеющее негативные последствия”, использовал в своих пьесах Шекспир!

Книжная полка

Книга «400 вопросов по информатике на логику и смекалку»

Издательство ДМК Пресс готовит к изданию книгу Д.М. Златопольского «400 вопросов по информатике на логику и смекалку».

| | |
|--|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| | <p>В книге приведены более 400 вопросов по информатике, которые можно предложить учащимся при проведении классных и школьных конкурсов «Что? Где? Когда?» и «Брейн-ринг» и других внеклассных мероприятий. Ряд вопросов может быть использован и на уроках. Вопросы охватывают многие темы школьного курса информатики, имеют разный уровень сложности. В приложениях представлены другие вопросы и задания, связанные с интеллектуальными соревнованиями. Ко всем вопросам даны ответы.</p> <p>Книга предназначена педагогам, но может быть полезна также родителям учащихся и, конечно, самим учащимся, которые могут проверить свои знания, логику и смекалку.</p> |
|--|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

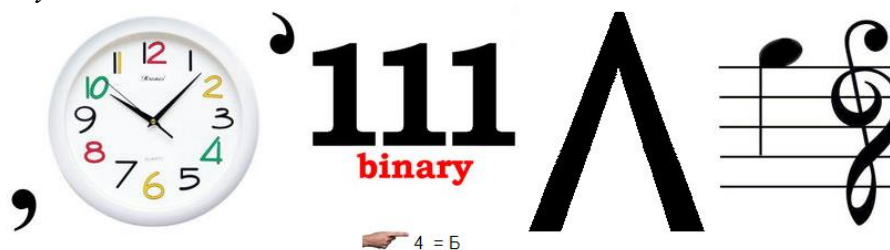
Пять ребусов

Решите, пожалуйста, ребусы.

Ребус 1



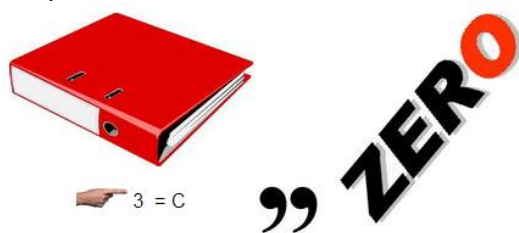
Ребус 2



Ребус 3



Ребус 4



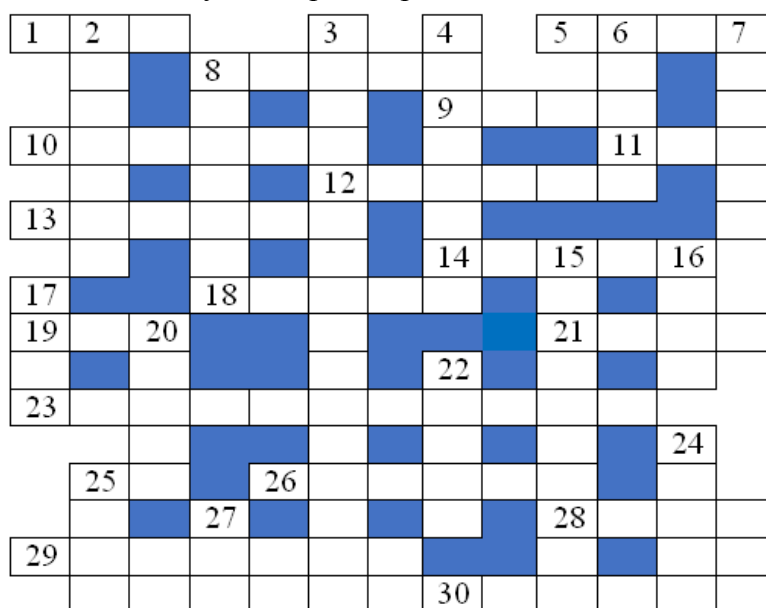
Ребус 5



Ответы присылайте в редакцию.

Кроссворд

Решите, пожалуйста, кроссворд.



По горизонтали

1. Характеристика файла или переменной величины.
5. Синоним слова “дорожка” (как части магнитного диска), происходящий от английского аналога этого слова.

8.



9.



10. Форма взаимодействия пользователя с компьютером.
11. ... координат.
12. Устранение неисправности компьютера.
13. Наука о законах и формах мышления.
14. Последовательность символов, предназначенная для чтения человеком.
18. Порядковое число.
19. То, без чего не может работать исправный и включенный в сеть настольный компьютер.
21. Устройство для соединения двух участков локальной сети, а также транспортное сооружение.
23. Создание второго экземпляра файла, символа, фрагмента текста.
25. Шестая нота.
26. Политическая организация, а также совокупность товаров.
28. Знак препинания.
29. Утверждение, используемое в условных операторах.
30. Веб-ресурс, предоставляющий множество сервисов и услуг: поиск в Интернете, новости и др.

По вертикали

2. Устройство для вывода информации в персональном компьютере.
3. Процесс составления программы.
4. Аргумент или результат процедуры.
6. Управляемая по программе и имитирующая действия человека машина.
7. Тип топологии локальной компьютерной сети.
8. 1 000 000.
15. Разновидность транслятора, которая “переводит” в машинные коды всю программу целиком, а затем исполняет их.
16. Проверка работоспособности компьютера.
17. Структура данных, в которых применен принцип “последним пришел – первым вышел”.
20. Второй экземпляр (файла и др.).
22. Совокупность точек графического изображения.
24. Внешнее очертание, наружный вид предмета, а также объект базы данных.
25. Часть рабочей книги программы Microsoft Excel или подобной.
27. Единица измерения скорости передачи данных.

Двенадцать спичек

Из 12 спичек нужно на столе составить фигуру, в которой было бы три одинаковых четырехугольника и два одинаковых треугольника. Как это сделать?

Числовой ребус “ТРОПКА – ДОРОЖКА”

Решите, пожалуйста, числовой ребус:

$$\text{ТРОПКА} + \text{ТРОПКА} + \text{ТРОПКА} = \text{ДОРОЖКА}$$

Как принято в таких головоломках, в нем одинаковыми буквами зашифрованы одинаковые цифры, разными буквами – разные цифры.

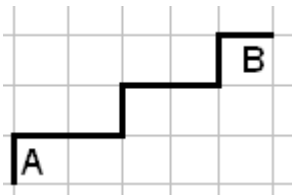
Головоломка “Цифры вместо «?»”

Какие цифры надо поставить вместо символов “?”, чтобы получить следующие равенства:

- 1) $(123\ 456\ 789 \times ?) + ? = 987\ 654\ 321$;
- 2) $(12\ 345\ 678 \times ?) + ? = 111\ 111\ 111$;
- 3) $12\ 345\ 679 \times ?? = 999\ 999\ 999$.

Задача “Лестница”

На координатной плоскости нужно нарисовать лестницу из трёх ступенек из точки А в точку В. Точка А имеет координаты (0, 0), а точка В — (5, 3). Каждая ступенька должна иметь одну единицу по высоте и целое количество единиц в длину. Один из возможных вариантов показан ниже:



Каждая лестница может быть закодирована тройкой чисел, задающих длины первой, второй и третьей ступеньки соответственно. Так, изображённая лесенка кодируется тройкой 2, 2, 1. Определить, сколько всего может быть таких лесенок, и перечислить все тройки чисел, соответствующие этим лесенкам.

Заполнение ячеек порядковыми номерами

При решении многих практических задач в среде программы Microsoft Excel или подобных часто требуется создать столбец или строку с порядковыми номерами. Это можно сделать разными способами.

Опишем их на примере диапазона ячеек A3:A100.

1. Автозаполнение

1. Ввести в ячейку A3 значение 1 (начальное), в ячейку A4 — 2.
2. Выделить эти ячейки.
3. Поставить курсор мыши на правый нижний угол прямоугольника, ограничивающего выделенные ячейки — курсор примет вид +. Эта так называемый “маркер заполнения”.
4. “Отбуксировать” мышью вниз на требуемое количество ячеек (при этом рядом с маркером появится маленький квадратик, в котором будет отображаться меняющееся значение текущего порядкового номера — *рис. 1*).

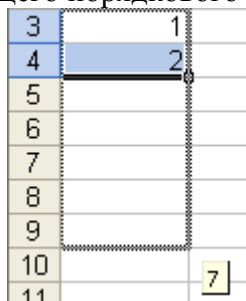


Рис. 1.1

Если вы отпустили кнопку мыши раньше, чем требуется, можно повторить описанные действия, используя на этапе 2 две последние заполненные ячейки.

Другой способ автозаполнения.


1. Ввести в ячейку A3 значение 1 (начальное).
2. Выделить ячейку A3.
3. Поставить курсор мыши на правый нижний угол прямоугольника, ограничивающего выделенные ячейки — появится маркер заполнения.
4. Нажать клавишу **Ctrl**, не отпуская ее, “отбуксировать” мышью вниз на требуемое количество ячеек.

2. Использование формул

1. В ячейку A3 ввести значение 1 (начальное).
2. В ячейку A4 ввести формулу =A3+1.
3. Используя маркер заполнения распространить (скопировать) формулу на остальные ячейки.

3. Использование прогрессии

В ячейке A3 набрать значение 1 (начальное) и ввести его, оставшись в этой же ячейке (для этого после набора нажать на клавишу **Enter** и щелкнуть мышью на кнопке с “галочкой”, на панели инструментов расположенной справа от адреса ячейки. Можно, конечно, после нажатия на клавишу **Enter**, вернуть указатель активной ячейки на A3.

Выбрать вкладку **Главная**, в группе Редактирование щелкнуть на пиктограмме **Заполнить** () и выбрать Прогрессия.

В появившемся диалоговом окне **Прогрессия** (рис. 1.2):

- в поле **Предельное значение** указать значение 98 (в данном случае);
- в группе радиокнопок **Расположение** указать значение по столбцам (в данном случае);
- нажать кнопку **ОК**.

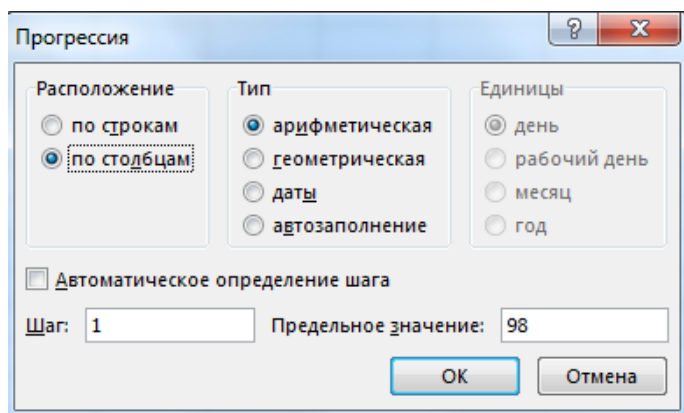


Рис. 2

Если перед вызовом меню выделить диапазон для ввода номеров, то предельное (последнее) значение задавать не нужно.

Другой способ использования прогрессии.

1. В ячейку A3 ввести значение 1 (начальное).
2. Поставить курсор мыши на маркер заполнения.
3. “Отбуксировать” мышь вниз на несколько ячеек, нажав ее правую кнопку.
4. После окончания выделения появится контекстное меню, в котором нужно выбрать пункт **Прогрессия** и в появившемся диалоговом окне с таким названием выполнить действия, описанные выше.

Если же в этом окне ничего не заполнять, а сразу щелкнуть на кнопке **ОК**, то порядковыми номерами будет заполнен выделенный диапазон.

Опишем также методику заполнения ячеек порядковыми номерами с точкой.

Пример. На листе представлена информация о результатах тестирования (рис. 3):

| | A | B | C | ... |
|----|--------------------------------|---------------|--------|-----|
| 1 | Результаты тестирования | | | |
| 2 | | Фамилия И.О. | Тест 1 | |
| 3 | 1. | Васильев Н.В. | | |
| 4 | 2. | Бойко К.С. | | |
| | | | | |
| 54 | 52. | Абрамов А.И. | | |

Рис. 3

Решение

1. Получить в диапазоне A3:A54 порядковые номера одним из описанных выше способов.
2. Выделить соответствующий диапазон.
3. Установить в нем так называемый “пользовательский формат”.

Для этого на вкладке **Главная**, в группе **Ячейки** выбрать пункт **Формат | Формат ячеек** и в появившемся диалоговом окне **Формат ячеек** (см. рис. 1.4) — вкладку **Число**, а в ней в списке **Числовые форматы** — пункт (все форматы).

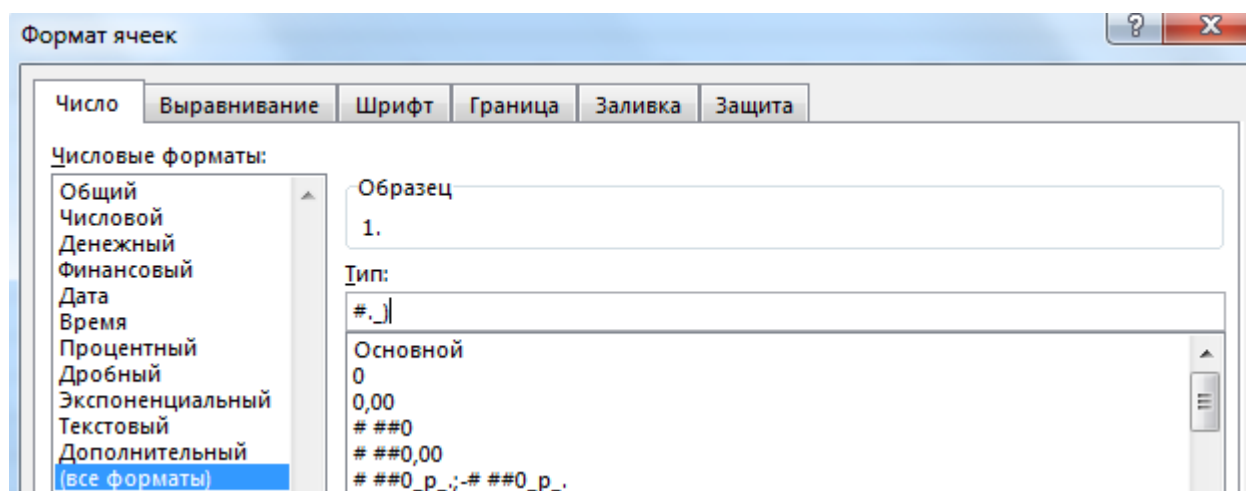


Рис. 4

В поле Тип в правой части окна ввести формат вида:

#.#)

и щелкнуть на кнопке **ОК**.

Диалоговое окно **Формат ячеек** можно также вызвать, набрав комбинацию клавиш **<Ctrl> + <1>**.

Поиск информации

Два вопроса

1. Эту породу собак англичане всегда считали признаком Германии. А потому в годы Первой мировой войны на туманном Альбионе считалось «весьма патриотичным» пинать их при первой возможности. Что это за порода?

2. Драматург Генрик Ибсен держал у себя на письменном столе в пустой банке весьма необычное «домашнее животное». Оно успокаивало сочинителя своим видом во время творческого процесса. Кто там жил?

Внимание! Конкурс!

Конкурс № 28

(задания конкурса предназначены для учащихся 5–7-х классов)

Решите, пожалуйста, две задачи.

1. Имеются двое песочных часов: на 7 минуты и на 11 минут. Яйцо варится непрерывно 15 минут. Как отмерить это время при помощи имеющихся весов? Время, затрачиваемое на переворачивание часов, не учитывать.

2. Есть двое песочных часов: на 3 минуты и на 8 минут. Для приготовления эликсира бессмертия его надо непрерывно готовить ровно 7 минут. Как это сделать? Время, затрачиваемое на переворачивание часов, не учитывать.

Конкурс № 29

В качестве задания этого конкурса предлагаем выполнить задания (можно не все) для самостоятельной работы, предложенные в статье «Методика выполнения задания 5 демонстрационного варианта ЕГЭ по информатике и ИКТ 2021 года».

Ответы направьте в редакцию по адресу mirinfo@infojournal.ru. Пожалуйста, укажите в письме свои фамилию и имя, населённый пункт и номер школы, а также фамилию, имя и отчество учителя информатики. Срок представления ответов — 31 октября 2020 г.