

Методические рекомендации при подготовке учащихся к государственной итоговой аттестации в форме ЕГЭ по информатике и ИКТ

Занятие № 2 практико-ориентированного семинара 03 марта 2017 года

Решение заданий ЕГЭ по информатике и ИКТ повышенного уровня

Задания 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 24, демоверсия ЕГЭ-2017

Для успешной подготовки к экзаменам рекомендую использовать ресурсы сети Интернет на которых всегда имеется актуальная информация по вопросам ЕГЭ по информатике и ИКТ. Конкретно:

<http://ege.edu.ru/ru/> - официальный портал ЕГЭ;

<http://fipi.ru/> - портал федеральной службы по надзору в сфере образования и науки;

<http://kpolyakov.spb.ru/> - портал К.Ю. Поляков с материалами заданий, включающими необходимый теоретический материал по всем типам заданий, образцами решений заданий каждого типа, генератором заданий с выбором их набора и ответами, онлайн-тестами по каждому типу заданий, дополнительным методическим и образовательным материалом по информатике и ИКТ;

<https://inf-ege.sdangia.ru/> - портал для подготовки к ЕГЭ, позволяющий при выполнении заданий учащемуся выполнять самоконтроль, получать образец верного решения задания. Для учителя данный портал полезен тем, что его ресурсы позволяют учителю автоматизировать работу не только по подбору заданий, но и автоматизировать работу по проверке заданий. Для эффективного использования данного ресурса, необходима регистрация учителя и учащихся на портале.

<https://ege.yandex.ru/informatics/> - материалы известного поискового портала yandex.ru, содержит материалы для тренировки, при авторизации позволяет вести личную статистику.

Стратегией подготовки учащихся к экзамену является обеспечение успешной сдачи экзамена. Понятно, что для каждого учащихся успешность означает различный максимум. Поэтому учителю важно вести учёт достижений каждого учащегося и направлять его деятельность для достижения как можно более высоких результатов. Тактически верным будет поступательный путь от минимума к максимуму.

На втором семинаре рассматриваются задания повышенного уровня сложности (13 - 22, 24) из демоверсии ЕГЭ-2017, подходы к их решению.

Задание 13

Вычисление количества информации

При регистрации в компьютерной системе каждому пользователю выдаётся пароль, состоящий из 9 символов. Из соображений информационной безопасности каждый пароль должен содержать хотя бы 1 десятичную цифру, как прописные, так и строчные латинские буквы, а также не менее 1 символа из 6-символьного набора: «&», «#», «\$», «*», «!», «@». В базе данных для хранения сведений о каждом пользователе отведено одинаковое и минимально возможное целое число байт. При этом используют посимвольное кодирование паролей, все символы кодируют одинаковым и минимально возможным количеством бит. Кроме собственно пароля, для каждого пользователя в системе хранятся дополнительные сведения, для чего выделено целое число байт; это число одно и то же для всех пользователей.

Для хранения сведений о 20 пользователях потребовалось 500 байт. Сколько байт выделено для хранения дополнительных сведений об одном пользователе? В ответе запишите только целое число – количество байт.

Примечание. В латинском алфавите 26 букв.

Найдем информационный объём одного символа пароля. Из условия задачи известно, что для формирования пароля используются 10 цифр, 26 прописных, 26 строчных и 6 специальных символов. Тогда $N = 10 + 26 + 26 + 6 = 68$. Так как для одного символа пароля используется минимально возможное количество бит, то $68 \approx 128 = 2^7$, откуда $i = 7$ бит – информационный объём одного символа пароля. По условию задачи пароль состоит из 9 знаков, тогда информационный объём пароля в битах составляет $7 \cdot 9 = 63$ бита. Так как пароль занимает целое минимально возможное количество байт, то $63 / 8 = 7,875$ байт ≈ 8 байт. Тогда 20 паролей составляют $8 \cdot 20 = 160$ байт. Теперь найдём количество байт, отведённое для хранения дополнительных сведений всех пользователей. $500 - 160 = 340$ байт. Наконец, вычислим информационный объём, отведённый для хранения дополнительных сведений для одного пользователя. $340 / 20 = 17$ байт.

Ответ: 17.

Задание 14

Выполнение алгоритмов для исполнителя

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 69 идущих подряд цифр 8? В ответе запишите полученную строку.

НАЧАЛО

ПОКА **нашлось** (3333) ИЛИ **нашлось** (8888)

ЕСЛИ **нашлось** (3333)

ТО **заменить** (3333, 88)

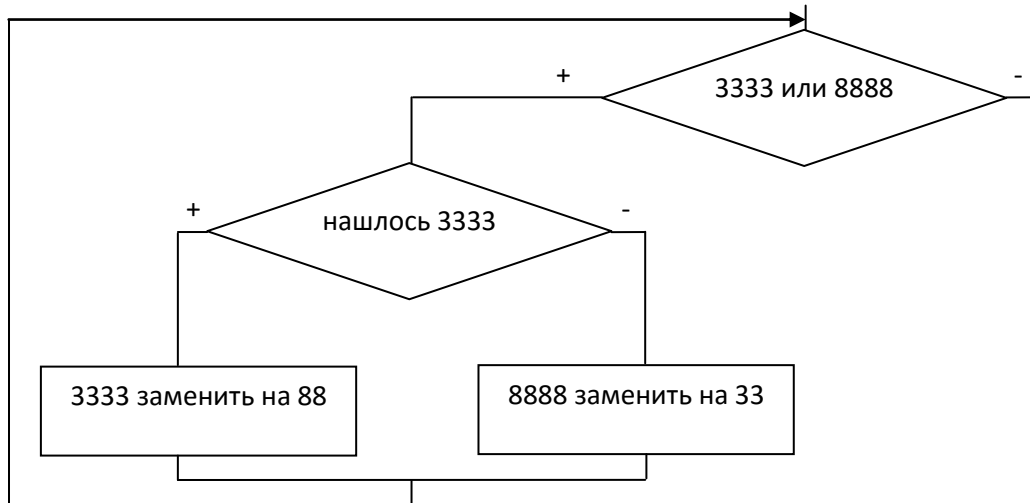
ИНАЧЕ **заменить** (8888, 33)

КОНЕЦ ЕСЛИ

КОНЕЦ ПОКА

КОНЕЦ

При выполнении данного задания уместно нарисовать блок-схему.



Анализируя алгоритм, приходим к выводу, что при троекратном выполнении цикла восемь цифр 8 заменяются на 88, следовательно, количество цифр 8 в последовательности уменьшается на шесть.

Исходя из условия задачи, что последовательность состоит из 69 цифр 8, имеем:

$$69 / 6 = 11 \text{ ост. } 3. \text{ Исходная последовательность уменьшилась на } 6 \cdot 11 = 66 \text{ цифр } 8.$$

Значит, останется три цифры 8.

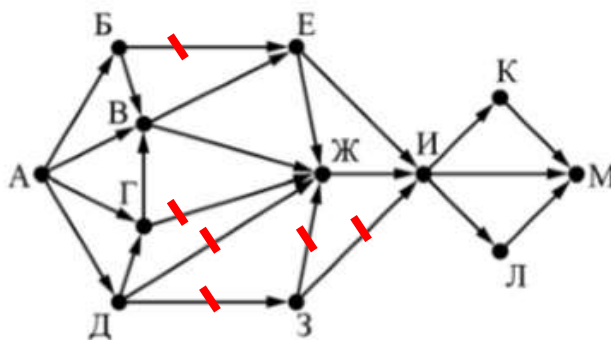
Ответ: 888.

Задание 15

Поиск путей в графе

На рисунке представлена схема дорог, связывающих города А, Б, В, Г, Д, Е, Ж, З, И, К, Л, М. По каждой дороге можно двигаться только в одном направлении, указанном стрелкой.

Сколько существует различных путей из города А в город М, проходящих через город В?



Так как по условию задачи все пути должны проходить обязательно через город В, то вычеркнем на схеме все пути, которые проходят не через город В.

Составим таблицу путей, ведущих в каждый город, до города И включительно. Так как из города И в город М количество путей равно 3, то, найденное количество путей, до И надо умножить на 3, что и будет количеством путей, ведущих в город М.

		Количество путей, ведущих в каждый город
Б	←А	1
В	←АБГ	1 + 1 + 2 = 4
Г	←АД	1 + 1 = 2
Д	←А	1

Задание 18

Проверка истинности логического выражения

Обозначим через $m \& n$ поразрядную конъюнкцию неотрицательных целых чисел m и n . Так, например, $14 \& 5 = 1110_2 \& 0101_2 = 0100_2 = 4_{10}$. Для какого наименьшего неотрицательного целого числа A формула

$$x \& 51 = 0 \vee (x \& 41 = 0 \rightarrow x \& A \neq 0)$$

тождественно истинна (т.е. принимает значение 1 при любом неотрицательном целом значении переменной x)?

$$51_{10} = 110011_2, 41_{10} = 101001_2$$

Воспользуемся табличным методом, с анализом каждого бита искомой величины A .

	$2^5 = 32$ 100000_2	$2^4 = 16$ 10000_2	$2^3 = 8$ 1000_2	$2^2 = 4$ 100_2	$2^1 = 2$ 10_2	$2^0 = 1$ 1_2
51	1	1	0	0	1	1
41	1	0	1	0	0	1
$x \& 51 = 0$	0	0	1	1	0	0
$x \& 41 = 0$	0	1	0	1	1	0
$x \& A$	*	1	*	*	1	*
	0	1	0	0	1	0

Знак * означает, что здесь может быть любой символ: 0 или 1.

Так как по условию задачи A - наименьшее, то вместо * берём 0.

$$16 + 2 = 18.$$

Ответ: 18.

Примечание.

$x \& 51 = 0$ означает, что в битах числа x необходимо обеспечить 0 там, где в битах числа 51 имеется 1.

Если бы в условии было $x \& 51 \neq 0$, то в битах числа x необходимо обеспечить 1 там, где в битах числа 51 имеется 1.

Задание 19

Обработка массивов и матриц

В программе используется одномерный целочисленный массив A с индексами от 0 до 9. Значения элементов равны 1, 2, 5, 8, 9, 3, 4, 0, 7, 6 соответственно, т.е. $A[0] = 1$, $A[1] = 2$ и т.д.

Определите значение переменной j после выполнения следующего фрагмента программы (записанного ниже на пяти языках программирования).

Бейсик	Python
<pre> j = 5 WHILE A(j) < A(j-1) t = A(j) A(j) = A(j-1) A(j-1) = t j = j - 1 WEND </pre>	<pre> j = 5 while A[j] < A[j-1]: A[j], A[j-1] = A[j-1], A[j] j -= 1 </pre>
Алгоритмический язык	Паскаль
<pre> j := 5 нц пока A[j] < A[j-1] t := A[j] A[j] := A[j-1] A[j-1] := t j := j - 1 кц </pre>	<pre> j := 5; while A[j] < A[j-1] do begin t := A[j]; A[j] := A[j-1]; A[j-1] := t; j := j - 1; end; </pre>

Анализируя фрагмент программы, приходим к выводу, что в массиве элементы, начиная с 5-го, меняются местами, если текущий элемент меньше предыдущего. Так как j уменьшается, то рассматриваются только элементы массива от 5 до 0.

j	Индексы	0	1	2	3	4	5	6	7	8	9		
5	Элементы	1	2	5	8	9	3	4	0	7	6		
4						3	9					$A[5] < A[4]$	Истина
3					3	8						$A[4] < A[3]$	Истина
2				3	5							$A[3] < A[2]$	Истина
												$A[2] < A[1]$	Ложь

Ответ: 2.

Задание 20

Анализ программ с циклами и условными операторами

Ниже на пяти языках программирования записан алгоритм. Получив на вход натуральное число x , этот алгоритм печатает число R . Укажите такое число x , при вводе которого алгоритм печатает двузначное число, сумма цифр которого равна 16. Если таких чисел x несколько, укажите наименьшее из них.

Бейсик	Python
<pre> DIM X,D,R AS LONG INPUT X R = 0 WHILE X>0 D = X MOD 10 R = 10*R + D X = X \ 10 WEND PRINT R </pre>	<pre> x = int(input()) R = 0 while x>0: d = x % 10 R = 10*R + d x = x // 10 print(R) </pre>
Алгоритмический язык	Паскаль
<pre> алг нач цел x, d, R ввод x R := 0 нц пока x>0 d := mod(x, 10) R := 10*R + d x := div(x, 10) кц вывод R кон </pre>	<pre> var x,d,R: longint; begin readln(x); R := 0; while x>0 do begin d := x mod 10; R := 10*R + d; x := x div 10 end; writeln(R) end. </pre>

Из анализа программы следует, что из введённого числа выделяется цифра единиц и присваивается переменной R . $R := 10*R + d$. Затем цифра единиц отбрасывается. Так как полученное число по условию задачи двузначное, а сумма его цифр равна 16, то цифрами исходного числа могут быть только 7 и 9 или 8 и 8. По условию задачи надо указать наименьшее число, следовательно, искомым числом является 79.

Ответ: 79.

Задание 21

Анализ программ с циклами и подпрограммами

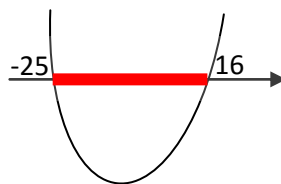
Напишите в ответе число, которое будет напечатано в результате выполнения следующего алгоритма (для Вашего удобства алгоритм представлен на пяти языках программирования).

Бейсик	Python
<pre>DIM A, B, N, t AS INTEGER A = -100: B = 100 N = 0 FOR t = A TO B IF F(t) <= 0 THEN N = N + 1 END IF NEXT t PRINT N FUNCTION F (x) F = (x - 16)*(x + 25) END FUNCTION</pre>	<pre>def f(x): return (x - 16)*(x + 25) a = -100 b = 100 n = 0 for t in range(a, b + 1): if f(t) <= 0: n = n + 1 print(n)</pre>
Алгоритмический язык	Паскаль
<pre>алг нач цел a, b, N, t a := -100; b := 100 N := 0 нц для t от a до b если F(t) <= 0 то N := N + 1 все кц вывод N кон алг цел F(цел x) нач знач := (x - 16)*(x + 25) кон</pre>	<pre>var a, b, N, t: integer; Function F(x: integer):integer; begin F := (x - 16)*(x + 25) end; begin a := -100; b := 100; N := 0; for t := a to b do begin if (F(t) <= 0) then N := N + 1 end; write(N) end.</pre>

Анализируя программу, приходим к выводу, что процедура-функция вычисляет значения квадратичной функции на отрезке $[-100, 100]$ с шагом $+1$ и учитывает количество найденных значений, когда они отрицательны или ноль.

Следовательно, $N = 16 - (-25) + 1 = 42$.

Решение задачи иллюстрируем наглядно, используя график квадратичной функции.



Ответ: 42.

Задание 22

Перебор вариантов, динамическое программирование

Исполнитель A16 преобразует число, записанное на экране.

У исполнителя есть три команды, которым присвоены номера:

1. Прибавить 1
2. Прибавить 2
3. Умножить на 2

Первая из них увеличивает число на экране на 1, вторая увеличивает его на 2, третья умножает его на 2.

Программа для исполнителя A16 – это последовательность команд.

Сколько существует таких программ, которые исходное число 3 преобразуют в число 12 и при этом траектория вычислений программы содержит число 10?

Траектория вычислений программы – это последовательность результатов выполнения всех команд программы. Например, для программы 132 при исходном числе 7 траектория будет состоять из чисел 8, 16, 18.

$$F(n) = F(n-1) + F(n-2) + F(n/2) \text{ для } n \leq 10$$

$$F(11) = F(10)$$

$$F(12) = F(10) + F(11)$$

n	3	4	5	6	7	8	9	10	11	12
F(n)	1	1	2	4	6	11	17	30	30	60
			1+1	2+1+1	4+2	6+4+1	11+6	17+11+2	30	30+30

Ответ: 60.

Задание 24

Поиск ошибок в программе со сложным условием

Часть 2

Для записи ответов на задания этой части (24–27) используйте БЛАНК ОТВЕТОВ № 2. Запишите сначала номер задания (24, 25 и т. д.), а затем полное решение. Ответы записывайте четко и разборчиво.

Дано целое положительное число N , не превосходящее 1000. Необходимо определить, является ли это число степенью числа 3. То есть требуется определить, существует ли такое целое число K , что $3^K = N$, и вывести это число либо сообщение, что такого числа не существует.

Для решения этой задачи ученик написал программу, но, к сожалению, его программа оказалась неверной. Ниже эта написанная им программа для Вашего удобства приведена на пяти языках программирования.

Бейсик	Python
<pre> DIM N, K AS INTEGER INPUT N K = 0 WHILE K MOD 3 = 0 K = K + 1 N = N \ 3 WEND IF N > 0 THEN PRINT K ELSE PRINT "Не существует" END IF END </pre>	<pre> n = int(input()) k = 0 while k%3 == 0: k = k + 1 n = n // 3 if n > 0: print(k) else: print("Не существует") </pre>
Алгоритмический язык	Паскаль
<pre> алг нач цел n, k ввод n k := 0 нц пока mod(k, 3) = 0 k := k + 1 n := div(n, 3) кц если n > 0 то вывод k иначе вывод "Не существует" все кон </pre>	<pre> var n, k: integer; begin readln(n); k := 0; while k mod 3 = 0 do begin k := k + 1; n := n div 3; end; if n > 0 then writeln(k) else writeln('Не существует') end. </pre>

C#

```

#include <stdio.h>
int main() {
    int n, k;
    scanf("%d", &n);
    k = 0;
    while (k%3 == 0) {
        k = k + 1;
        n = n / 3;
    }
    if (n > 0)
        printf("%d", k);
    else
        printf("Не существует");
    return 0;
}
                    
```

Последовательно выполните следующее.

1. Напишите, что выведет эта программа при вводе числа 9.
2. Приведите пример числа, при вводе которого приведённая программа напечатает то, что требуется.
3. Найдите в программе все ошибки (их может быть одна или несколько). Для каждой ошибки выпишите строку, в которой она допущена, и приведите эту же строку в исправленном виде.

Достаточно указать ошибки и способ их исправления для одного языка программирования.

Обратите внимание: Вам нужно исправить приведённую программу, а не написать свою. Вы можете только заменять ошибочные строки, но не можете удалять строки или добавлять новые. Заменять следует только ошибочные строки: за исправления, внесённые в строки, не содержащие ошибок, баллы будут снижаться.

Выполняя постановку задачи, замечаем, что результатом работы программы должен быть либо показатель степени числа 3, если введённое число есть 3^k , либо "не существует",

если введённое число не является степенью числа 3. Если n является степенью числа 3, то программа прекращает цикл, когда n становится равным 1. Если число не является степенью числа 2, то цикл не выполняется ни разу.

Важно заметить, что при ответе на поставленные вопросы, не требуется никаких пояснений, почему программа выводит тот или иной результат. Нужно выполнить конкретные задания и строго ответить на вопросы.

Ответ на первый вопрос.

При вводе числа 9 программа выведет 1.

Ответ на второй вопрос.

При вводе числа 3 программа выведет 1, что является верным ответом.

Ответ на третий вопрос.

В программе допущены две ошибки.

Первая ошибка: `while k mod 3 = 0 do begin`

Исправленная ошибка: `while n mod 3 = 0 do begin`

Вторая ошибка: `if n>0 then`

Исправленная ошибка: `if n=1 then`