

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Смоленский государственный университет»

**ПРОГРАММА ВСТУПИТЕЛЬНОГО ИСПЫТАНИЯ
ПО ОСНОВАМ ПРОГРАММИРОВАНИЯ**

Смоленск

СОДЕРЖАНИЕ ОСНОВНЫХ РАЗДЕЛОВ

Для успешной сдачи экзамена необходимо владеть следующими знаниями и умениями:

1. Умение представлять и считывать данные в разных типах информационных моделей (схемы, карты, таблицы, графики и формулы).
2. Умение кодировать и декодировать информацию.
3. Формальное исполнение алгоритма, записанного на естественном языке, или умение создавать линейный алгоритм для формального исполнителя с ограниченным набором команд.
4. Знание основных конструкций языка программирования, понятия переменной, оператора присваивания.
5. Умение анализировать результат исполнения алгоритма.
6. Вычисление рекуррентных выражений.
7. Умение создавать собственные программы (20–40 строк) для обработки целочисленной информации.
8. Умение анализировать алгоритм, содержащий ветвление и цикл.
9. Умение анализировать результат исполнения алгоритма.
10. Умение создавать собственные программы (10–20 строк) для обработки символьной информации.
11. Умение создавать собственные программы (10–20 строк) для обработки целочисленной информации.

Ниже приводится возможный вариант экзаменационной работы (задачи взяты из демоверсии ЕГЭ 2021 от ФИПИ).

1. Для кодирования некоторой последовательности, состоящей из букв Л, М, Н, П, Р, решили использовать неравномерный двоичный код, удовлетворяющий условию, что никакое кодовое слово не является началом другого кодового слова. Это условие обеспечивает возможность однозначной расшифровки закодированных сообщений. Для букв Л, М, Н использовали соответственно кодовые слова 00, 01, 11. Для двух оставшихся букв – П и Р – кодовые слова неизвестны.

Укажите кратчайшее возможное кодовое слово для буквы П, при котором код будет удовлетворять указанному условию. Если таких кодов несколько, укажите код с наименьшим числовым значением.

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

3. Определите, при каком наибольшем введённом значении переменной s программа выведет число 64. Для Вашего удобства программа представлена на четырёх языках программирования.

4. Игорь написал программу, которая обрабатывает сообщения, состоящие из букв Ш, К, Л. Программа выводит количество сообщений, в которых встречается буква Л. Игорь хочет, чтобы программа вывела 64. Для Вашего удобства программа представлена на четырёх языках программирования.

Паскаль	Python	Алгоритмический язык
<pre>var s, n: integer; begin readln (s); s := s div 10; n := 1; while s < 51 do begin s := s + 5; n := n * 2 end; writeln(n) end.</pre>	<pre>s = int(input()) s = s // 10 n = 1 while s < 51: s = s + 5 n = n * 2 print(n)</pre>	<pre>алг нач цел n, s ввод s s := div(s, 10) n := 1 нц пока s < 51 s := s + 5 n := n * 2 кц вывод n кон</pre>

ни сообщений, в количестве кодовых слов. Буква Л может быть только в сообщении. Каждая из других букв встречается ровно 1 раз или 2 раза. Игорь хочет использовать программу, которая выведет 64.

5. Игорь написал программу, которая преобразовывает сообщение, состоящее из букв Ш, К, Л. Программа выводит количество сообщений, в которых встречается буква Л. Игорь хочет, чтобы программа вывела 64. Для Вашего удобства программа представлена на четырёх языках программирования.

C++
<pre>#include <iostream> using namespace std; int main() { int s, n; cin >> s; s = s / 10; n = 1; while (s < 51) { s = s + 5; n = n * 2; } cout << n << endl; return 0; }</pre>

каждому сообщению. Буква Л встречается в сообщении. Каждая из других букв встречается ровно 1 раз или 2 раза. Игорь хочет использовать программу, которая выведет 64.

первое слева слово заменить на второе слово. Если первое слово встречается в сообщении, то второе слово не встречается.

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)

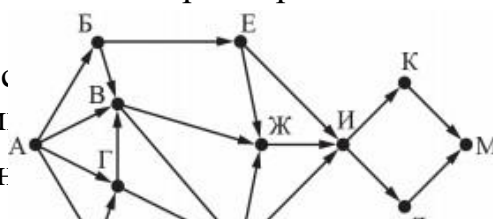
ИНАЧЕ **заменить** (8888, 22)

КОНЕЦ ЕСЛИ

КОНЕЦ ПОКА КОНЕЦ

6. 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?

7. Алгоритм вычисления значения функции $F(n)$, где n – натуральное число, задан следующим образом:



1), где n – натуральное

$$F(n) = 1 \text{ при } n = 1;$$

8. Рассматривается отрезок $[1016; 7937]$, количество таких чисел...

$$F(n) = n + F(n - 1), \text{ если } n - \text{ чётно,}$$

$$F(n) = 2 \times F(n - 2), \text{ если } n > 1 \text{ и при этом } n - \text{ нечётно.}$$

их числовому 27. Найдите

9. Ниже на четырёх языках программирования записан алгоритм. Получив на вход число x , этот алгоритм печатает два числа: L и M . Укажите наибольшее число x , при вводе которого алгоритм печатает сначала 4, а потом 5.

10. Исполните

C++	Python
<pre>#include <iostream> using namespace std; int main() { int x, L, M, Q; cin >> x; Q = 9; L = 0; while (x >= Q) { L = L + 1; x = x - Q; } M = x; cout << M << endl; }</pre>	<pre>x = int(input()) Q = 9 L = 0 while x >= Q: L = L + 1 x = x - Q M = x if M < L: M = L L = x print(L) print(M)</pre>

исполнителя есть две

команды, которым присвоены номера:

1. Прибавить 1
2. Умножить на 2

Первая команда увеличивает число на экране на 1, вторая умножает его на 2. Программа для исполнителя – это последовательность команд. Сколько существует программ, для которых при исходном числе 1 результатом является число 20, и при этом траектория вычислений содержит число 10?

Траектория вычислений программы – это последовательность результатов выполнения всех команд программы. Например, для программы 121 при исходном числе 7 траектория будет состоять из чисел 8, 16, 17.

11. Текстовый файл состоит не более чем из 106 символов X, Y и Z. Определите максимальное количество идущих подряд символов, среди которых каждые два соседних различны. Для выполнения этого задания следует написать программу.

12. Напишите программу, которая ищет среди целых чисел, принадлежащих числовому отрезку [174457; 174505], числа, имеющие ровно два различных натуральных делителя, не считая единицы и самого числа. Для каждого найденного числа запишите эти два делителя в таблицу на экране с новой строки в порядке возрастания произведения этих двух делителей. Делители в строке таблицы также должны следовать в порядке возрастания.

Например, в диапазоне [5; 9] ровно два целых различных натуральных делителя имеют числа 6 и 8, поэтому для этого диапазона таблица на экране должна содержать следующие значения:

2	3
2	4

Система оценивания экзаменационной работы по основам программирования

Всего заданий 12.

За правильный ответ на задания 1-10 ставится 1 первичный балл; за неверный ответ или его отсутствие – 0 баллов. За правильный ответ на задание 11 ставится 2 первичных балла, за задание 12 – 3 первичных балла, за неверный ответ или его отсутствие – 0 баллов.

Максимальный первичный балл за работу – 15. Итоговая оценка получается из первичного балла умножением баллов за задания 1-10 на 7, за задания 11 и 12 – на 6. Максимальная возможная оценка за работу 100 баллов.

Общее время выполнения работы – 180 минут.

Таблица. Ответы на задания

Номер задания	Правильный ответ	
1	100	
2	19	
3	259	
4	48	
5	22	
6	24	
7	4122	
8	1568	
9	49	
10	28	
11	35	
12	3	58153
	7	24923
	59	2957
	13	13421
	149	1171
	5	34897
	211	827
	2	87251

Список литературы

1. Информатика, Единый Государственный Экзамен, готовимся к итоговой аттестации, Лещинер В.Р., Крылов С.С., 2021.

2. Информатика и ИКТ, подготовка к ЕГЭ в 2021 году, диагностические работы, Зайдельман Я.Н., 2021.