

**Оборонно-техническая олимпиада 2022-2023 гг. (II этап)**

**Типовое задание для 8 класса**

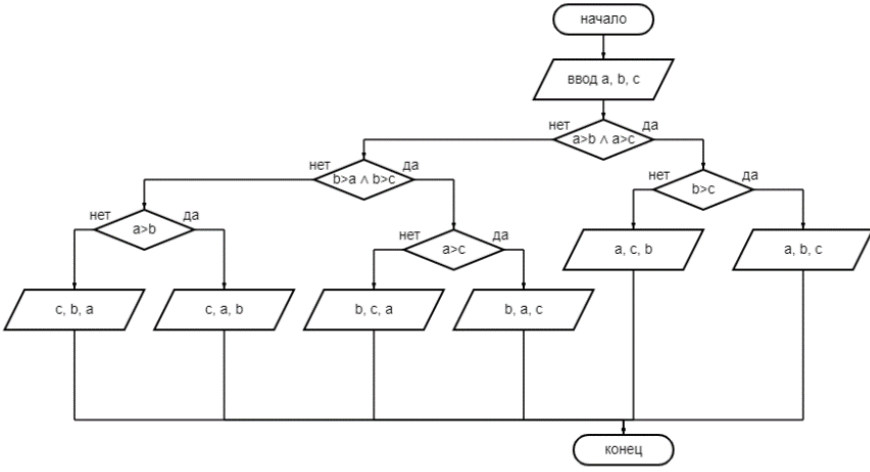
**Направление: Информатика и программирование**

1	Перевести в десятичную систему счисления числа: 110112, 1123678, AAF16.
2	Перевести в двоичную систему счисления числа: 4510; 1223, 21,310.
3	Рисунок размером 128 на 256 пикселей занимает в памяти 32 Кбайтов (без учёта сжатия). Найти максимально возможное количество цветов в палитре изображения.
4	Сколько значащих нулей в двоичной записи шестнадцатеричного числа 75BD16.
5	Составить и нарисовать блок схему алгоритма для представленной ниже задачи Ввести три числа и вывести их в порядке убывания.
6	Дан массив С из k элементов ( $k \leq 20$ ). Найти число элементов массива, не лежащих в интервале от -А до А (А и k вводятся с клавиатуры).
7	Найти сумму всех двухзначных чисел, кратных заданному натуральному k
8	Дан массив К, содержащий не более 15 элементов. Найти максимальный элемент массива и поменять его местами с первым элементом.
9	Дано четырехзначное число, сравнить первую и последнюю цифру данного числа.
10	Одноклеточная амеба каждые 4 часа делится на 2 клетки. Считая, что первоначально в замкнутом объеме находится 8 клеток, определить, через какое время в этом объеме будет находиться 120 клеток.

## Решение

### 8 класс

1	$110112 = 1 \cdot 24 + 1 \cdot 23 + 0 \cdot 22 + 1 \cdot 21 + 1 \cdot 20 = 16 + 8 + 0 + 2 + 1 = 2710$ $1123678 = 1 \cdot 85 + 1 \cdot 84 + 2 \cdot 83 + 3 \cdot 82 + 6 \cdot 81 + 7 \cdot 80 =$ $32768 + 4096 + 1024 + 192 + 48 + 7 = 3813510$ $AAF16 = 10 \cdot 162 + 10 \cdot 161 + 15 \cdot 160 = 2560 + 160 + 15 = 273510$	5
2	<div style="margin-bottom: 10px;"> <math display="block">  \begin{array}{r}  45 \overline{) 2} \\  \underline{-44} \phantom{0} \\  1 \phantom{0} \overline{) 22} \phantom{0} \\  \underline{-22} \phantom{00} \\  0 \phantom{0} \overline{) 11} \phantom{0} \\  \phantom{0} \overline{-10} \phantom{0} \overline{) 5} \phantom{0} \\  \phantom{0} \phantom{0} \overline{-4} \phantom{0} \overline{) 2} \phantom{0} \\  \phantom{0} \phantom{0} \phantom{0} \overline{-2} \phantom{0} \overline{) 1} \phantom{0} \\  \phantom{0} \phantom{0} \phantom{0} \phantom{0} \overline{0}  \end{array}  </math> </div> <p>4510 = 1011012</p> <div style="margin-bottom: 10px;"> <math display="block">  \begin{array}{r}  17 \overline{) 2} \\  \underline{-16} \phantom{0} \\  1 \phantom{0} \overline{) 8} \phantom{0} \\  \underline{-8} \phantom{00} \\  0 \phantom{0} \overline{) 4} \phantom{0} \\  \phantom{0} \overline{-4} \phantom{0} \overline{) 2} \phantom{0} \\  \phantom{0} \phantom{0} \overline{0} \phantom{0} \overline{) 2} \phantom{0} \\  \phantom{0} \phantom{0} \phantom{0} \overline{-2} \phantom{0} \overline{) 1} \phantom{0} \\  \phantom{0} \phantom{0} \phantom{0} \phantom{0} \overline{0}  \end{array}  </math> </div> <p>1710 = 100012</p> <p>Целая часть:</p> <div style="margin-bottom: 10px;"> <math display="block">  \begin{array}{r}  21 \overline{) 2} \\  \underline{-20} \phantom{0} \\  1 \phantom{0} \overline{) 10} \phantom{0} \\  \underline{-10} \phantom{00} \\  0 \phantom{0} \overline{) 5} \phantom{0} \\  \phantom{0} \overline{-4} \phantom{0} \overline{) 2} \phantom{0} \\  \phantom{0} \phantom{0} \overline{-1} \phantom{0} \overline{) 2} \phantom{0} \\  \phantom{0} \phantom{0} \phantom{0} \overline{-2} \phantom{0} \overline{) 1} \phantom{0} \\  \phantom{0} \phantom{0} \phantom{0} \phantom{0} \overline{0}  \end{array}  </math> </div> <p>Получилось: 2110 = 101012</p> <p>Дробная часть числа находится умножением на основание новой</p>	5

	<table border="1"> <tbody> <tr><td>0</td><td>3</td></tr> <tr><td>.</td><td>2</td></tr> <tr><td>0</td><td>6</td></tr> <tr><td></td><td>2</td></tr> <tr><td>1</td><td>2</td></tr> <tr><td></td><td>2</td></tr> <tr><td>0</td><td>4</td></tr> <tr><td></td><td>2</td></tr> <tr><td>0</td><td>8</td></tr> <tr><td></td><td>2</td></tr> <tr><td>1</td><td>6</td></tr> <tr><td></td><td>2</td></tr> <tr><td>1</td><td>2</td></tr> <tr><td></td><td>2</td></tr> <tr><td>0</td><td>4</td></tr> <tr><td></td><td>2</td></tr> <tr><td>0</td><td>8</td></tr> <tr><td></td><td>2</td></tr> <tr><td>1</td><td>6</td></tr> <tr><td></td><td>2</td></tr> <tr><td>1</td><td>2</td></tr> <tr><td></td><td>2</td></tr> <tr><td>0</td><td>4</td></tr> <tr><td></td><td>2</td></tr> </tbody> </table>	0	3	.	2	0	6		2	1	2		2	0	4		2	0	8		2	1	6		2	1	2		2	0	4		2	0	8		2	1	6		2	1	2		2	0	4		2	
0	3																																																	
.	2																																																	
0	6																																																	
	2																																																	
1	2																																																	
	2																																																	
0	4																																																	
	2																																																	
0	8																																																	
	2																																																	
1	6																																																	
	2																																																	
1	2																																																	
	2																																																	
0	4																																																	
	2																																																	
0	8																																																	
	2																																																	
1	6																																																	
	2																																																	
1	2																																																	
	2																																																	
0	4																																																	
	2																																																	
3	<p>Необходимо рассчитать информационный объем одного пикселя для этого используем формулу <math>N=2I</math>, где <math>N</math> – количество цветов в палитре, а <math>I</math> – количество информации на один цвет или пиксель. Выразим <math>I</math> из формулы <math>V = x * y * I</math>, где <math>V</math> - объем, <math>x</math> - ширина изображения (в нашем случае 128 пикселей), <math>y</math> - высота изображения (в нашем случае 256 пикселей).</p> $I = V / x * y$ $I = V / 128 \times 256$ <p><math>V</math> необходимо перевести в биты. 32 килобайта = <math>32 * 8 * 1024</math> бит</p> $I = 32 * 8 * 1024 / 128 * 256 = 32 * 23 * 210 / 27 * 28 = 32 / 4 = 8 \text{ бит.}$ <p><math>N = 2I = 256</math> цветов.</p> <p>Ответ: 256 цветов.</p>	5																																																
4	<p><math>75BD_{16} = 111010110111101</math></p> <p>Незначащие нули — это нули, стоящие слева от первой единицы и справа от последней единицы. Таких нет</p> <p>Ответ: 4 значащих нуля</p>	5																																																
5	 <pre> graph TD     Start([начало]) --&gt; Input[/ввод a, b, c/]     Input --&gt; D1{a &gt; b &amp; a &gt; c}     D1 -- да --&gt; Out1[/a, b, c/]     D1 -- нет --&gt; D2{b &gt; a &amp; b &gt; c}     D2 -- да --&gt; Out2[/b, c, a/]     D2 -- нет --&gt; D3{a &gt; c}     D3 -- да --&gt; Out3[/b, a, c/]     D3 -- нет --&gt; Out4[/c, b, a/]     Out1 --&gt; End([конец])     Out2 --&gt; End     Out3 --&gt; End     Out4 --&gt; End   </pre>	10																																																

6	<pre> # Запрашиваем значения k и A у пользователя k = int(input("Введите количество элементов массива: ")) A = int(input("Введите значение A: "))  # Создаем массив C C = [] for i in range(k):     element = int(input(f"Введите значение {i+1}-го элемента массива: "))     C.append(element)  # Находим число элементов, не лежащих в интервале от -A до A count = 0 for element in C:     if element &lt; -A or element &gt; A:         count += 1  # Выводим результат print(f"Число элементов массива, не лежащих в интервале от -{A} до {A}: {count}") </pre>	10
7	<pre> # Запрашиваем у пользователя натуральное число k k = int(input("Введите натуральное число k: ")) # Инициализируем счетчик суммы sum = 0 # Проходим по каждому двухзначному числу for i in range(10, 100):     # Если число кратно k, добавляем его к сумме     if i % k == 0:         sum += i # Выводим результат print(f"Сумма всех двухзначных чисел, кратных {k}: {sum}") </pre>	10

8	<pre># Задаем массив K K = [5, 7, 2, 9, 4, 8] # Находим максимальный элемент и его индекс в массиве K max_val = max(K) max_idx = K.index(max_val) # Меняем местами первый элемент и максимальный элемент K[0], K[max_idx] = K[max_idx], K[0] # Выводим измененный массив K print("Измененный массив K:", K)</pre>	15
9	<pre>number = int(input("Введите четырехзначное число: ")) first_digit = number // 1000 last_digit = number % 10  if first_digit == last_digit:     print("Первая и последняя цифры равны") else:     print("Первая и последняя цифры не равны")</pre>	15
10	<pre># Исходное количество клеток n = 8 # Целевое количество клеток m = 120 # Время, прошедшее с начала наблюдения (в часах) t = 0 # Пока количество клеток меньше целевого, продолжаем делить клетки каждые 4 часа while n &lt; m:     n *= 2     t += 4 # Выводим результат print("Через", t, "часов в замкнутом объеме будет находиться", n, "клеток.")</pre>	20

**Оборонно-техническая олимпиада 2022-2023 гг. (II этап)**

**Типовое задание для 9 класса**

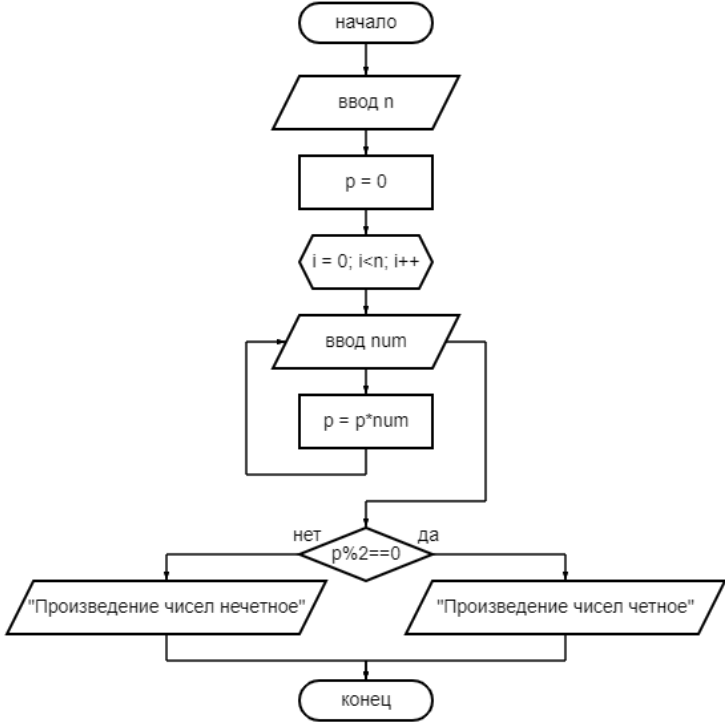
**Направление: Информатика и программирование**

1	Представить в формате с плавающей точкой (в четырехбайтовом формате) число 51,1.
2	Сколько значащих нулей в двоичной записи шестнадцатеричного числа 61A?
3	Какое максимальное количество целых страниц текста, содержащего 40 символов в строке и 20 строк на странице, использующего алфавит в 35000 символов, может содержать текстовый файл, сохраненный на гибком магнитном диске 2,5" ? Будем считать, что для хранения данных на таком диске выделяется 2400 секторов объемом 256 байтов.
4	Сколько значащих нулей в двоичной записи шестнадцатеричного числа 75BD16.
5	Составить и нарисовать блок схему в соответствии с задачей Определить, является ли произведение $N$ введенных натуральных чисел четной.
6	Ввести три числа и вывести их в порядке возрастания.
7	В каждом столбце заданной матрицы определить сумму четных значений.
8	Найти все трехзначные числа, которые являются простыми.
9	Дана матрица $M$ . Найти сумму элементов последней строки и заменить первый и третий элемент этой строки полученной суммой.
10	Дан массив целых чисел $C$ из $n$ элементов. Получить новый массив удалением из исходного массива всех положительных и нулевых элементов.

## Решение

### 9 класс

1	<p>Перевести число в двоичный вид: <math>51 = 110011</math> <math>0,1 = 0.00011</math> (перевод в двоичную дробь) <math>51,1 = 110011.00011</math></p>	5
2	<p>Для решения задачи нужно перевести шестнадцатеричное число 61A в двоичную систему счисления и посчитать количество значащих нулей. Переведем число 61A в двоичную систему счисления: <math>6 = 0110</math> <math>1 = 0001</math> <math>A = 1010</math> Таким образом, 61A в двоичной системе счисления будет выглядеть как 011000011010. Теперь посчитаем количество значащих нулей, то есть количество нулей между первой единицей слева и последней единицей справа. В данном случае между первой единицей слева и последней единицей справа находится 6 нулей. Ответ: 6.</p>	5
3	<p>Для решения задачи необходимо вычислить объем одной страницы текста и затем поделить общий объем диска на объем одной страницы. Объем одной страницы текста считается так: 40 символов на строку * 20 строк на страницу = 800 символов на страницу 800 символов на страницу * 2 байта на символ (для хранения символов в алфавите в 35000 символов) = 1600 байт на страницу Общий объем диска: 2400 секторов * 256 байтов на сектор = 614400 байтов Максимальное количество страниц текста: 614400 байтов / 1600 байтов на страницу = 384 страницы Ответ: максимальное количество целых страниц текста, которое может содержать текстовый файл, сохраненный на гибком магнитном диске 2,5” составляет 384 страницы.</p>	5
4	<p><math>75BD16=111010110111101</math> Незначащие нули — это нули, стоящие слева. Таких нет Ответ: 4 значащих нуля</p>	5

5	 <pre> graph TD     Start([начало]) --&gt; Input[/ввод n/]     Input --&gt; P0[p = 0]     P0 --&gt; Loop{i = 0; i &lt; n; i++}     Loop --&gt; InputNum[/ввод num/]     InputNum --&gt; Pmult[p = p * num]     Pmult --&gt; Loop     Loop --&gt; Decision{p % 2 == 0}     Decision -- нет --&gt; OutputOdd[/Произведение чисел нечетное/]     Decision -- да --&gt; OutputEven[/Произведение чисел четное/]     OutputOdd --&gt; End([конец])     OutputEven --&gt; End   </pre>	10
6	<pre> # Запрашиваем у пользователя три числа num1 = int(input("Введите первое число: ")) num2 = int(input("Введите второе число: ")) num3 = int(input("Введите третье число: ")) # Определяем максимальное и минимальное значения из трех чисел min_num = min(num1, num2, num3) max_num = max(num1, num2, num3) # Определяем среднее значение из трех чисел mid_num = (num1 + num2 + num3) - min_num - max_num # Выводим числа в порядке возрастания print(f"Числа в порядке возрастания: {min_num}, {mid_num}, {max_num}")   </pre>	10
7	<pre> # Задаем матрицу в виде списка списков matrix = [     [2, 3, 5, 8],     [1, 6, 7, 10],     [4, 9, 12, 15] ] # Определяем количество строк и столбцов в матрице rows = len(matrix) cols = len(matrix[0]) # Создаем список для хранения сумм четных значений в каждом   </pre>	10



	<pre> столбце even_sums = [0] * cols # Проходим по каждому столбцу матрицы for j in range(cols):     # Проходим по каждой строке столбца     for i in range(rows):         # Если элемент является четным, добавляем его к сумме         # четных значений столбца         if matrix[i][j] % 2 == 0:             even_sums[j] += matrix[i][j] # Выводим результат for j in range(cols):     print(f"Сумма четных значений в столбце {j+1}: {even_sums[j]}") </pre>	
8	<pre> # Функция для проверки, является ли число простым def is_prime(n):     if n &lt; 2:         return False     for i in range(2, int(n ** 0.5) + 1):         if n % i == 0:             return False     return True  # Список для хранения простых трехзначных чисел primes = [] # Перебираем все трехзначные числа for num in range(100, 1000):     # Если число простое, добавляем его в список     if is_prime(num):         primes.append(num) # Выводим список простых трехзначных чисел print(primes) </pre>	15
9	<pre> C = [3, -2, 0, 5, -1, 7, 0, -4] new_C = [] for num in C:     if num &lt;= 0:         new_C.append(num) print(new_C) </pre>	15
10	<pre> # Создаем матрицу M M = [[1, 2, 3], </pre>	20

```
[4, 5, 6],  
[7, 8, 9],  
[10, 11, 12],  
[13, 14, 15]]
```

```
# Находим сумму элементов последней строки
```

```
last_row = M[-1]
```

```
sum_last_row = sum(last_row)
```

```
# Заменяем первый и третий элемент последней строки на сумму  
её элементов
```

```
last_row[0] = sum_last_row
```

```
last_row[2] = sum_last_row
```

```
# Выводим полученную матрицу на экран
```

```
for row in M:
```

```
    print(row)
```

**Оборонно-техническая олимпиада 2022-2023 гг. (II этап)**

**Типовое задание для 10 класса**

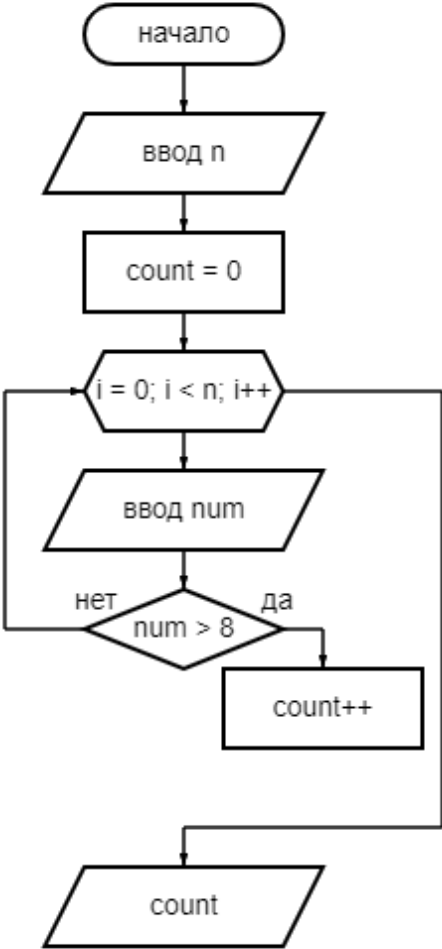
**Направление: Информатика и программирование**

1	Перевести в двоичную систему счисления числа: 5810; 1408; В316.
2	Представить в формате с фиксированной точкой (в однобайтовом формате со знаком) число -80 в прямом коде
3	Рисунок размером 128 на 128 пикселей занимает в памяти 250 Кбайтов (без учёта сжатия). Найти максимально возможное количество цветов в палитре изображения.
4	Производится четырёхканальная (квадро) звукозапись с частотой дискретизации 16 кГц и 16- битным разрешением. Запись длится 1.5 минуты, её результаты записываются в файл, сжатие данных не производится. Определить приблизительно размер полученного файла (в мегабайтах)
5	Составить и нарисовать блок схему в соответствии с задачей для 10 класса С клавиатуры вводится последовательность из N чисел. Определить, сколько из них больше восьми.
6	Вычислить $N=1!+2!+\dots+n!$ . Значение n вводить с клавиатуры.
7	Дан одномерный массив K из n элементов. Найти среднее арифметическое нечетных элементов в массиве
8	Дана матрица целых чисел. В каждой строке найти среднее арифметическое элементов с нечетными индексами.
9	Проверить, является ли натуральное число n степенью числа 5. Операцию возведения в степень не использовать.
10	Дан массив N из k элементов и число P. Вычислить среднее арифметическое чисел, меньших P.

**Решение**  
**10 класс**

1	<p>Переводим целую часть 5810 в 2-ую систему последовательным делением на 2:</p> $58/2 = 29, \text{ остаток: } 0$ $29/2 = 14, \text{ остаток: } 1$ $14/2 = 7, \text{ остаток: } 0$ $7/2 = 3, \text{ остаток: } 1$ $3/2 = 1, \text{ остаток: } 1$ $1/2 = 0, \text{ остаток: } 1$ $5810 = 1110102$ <p>Переводим 1408 в десятичную систему:</p> $1408 = 1 \cdot 82 + 4 \cdot 81 + 0 \cdot 1 = 9610$ <p>Переводим целую часть 9610 в 2-ую систему последовательным делением на 2:</p> $96/2 = 48, \text{ остаток: } 0$ $48/2 = 24, \text{ остаток: } 0$ $24/2 = 12, \text{ остаток: } 0$ $12/2 = 6, \text{ остаток: } 0$ $6/2 = 3, \text{ остаток: } 0$ $3/2 = 1, \text{ остаток: } 1$ $1/2 = 0, \text{ остаток: } 1$ $9610 = 11000002$ <p>Переводим В316 в десятичную систему:</p> $В316 = 11 \cdot 161 + 3 \cdot 1 = 17910$ <p>Переводим целую часть 17910 в 2-ую систему последовательным делением на 2:</p> $179/2 = 89, \text{ остаток: } 1$ $89/2 = 44, \text{ остаток: } 1$ $44/2 = 22, \text{ остаток: } 0$ $22/2 = 11, \text{ остаток: } 0$ $11/2 = 5, \text{ остаток: } 1$ $5/2 = 2, \text{ остаток: } 1$ $2/2 = 1, \text{ остаток: } 0$ $1/2 = 0, \text{ остаток: } 1$ $17910 = 101100112$	5
2	<p>Для представления числа -80 в формате с фиксированной точкой в однобайтовом формате со знаком, сначала нужно определить количество битов, занимаемых целой и дробной частями числа.</p>	5

	<p>Пусть целая часть занимает 6 битов, а дробная часть - 2 бита. Тогда необходимо перевести модуль числа 80 в двоичную систему счисления и разделить его на две части: первые 6 цифр - это целая часть числа, оставшиеся 2 цифры - дробная часть.</p> $80 = 1010000_2$ <p>Целая часть: <math>101000_2 = 40_{10} = 00101000_2</math>  Дробная часть: <math>00_2</math></p> <p>Знак числа -80 - отрицательный, поэтому необходимо выполнить дополнение до двойки для целой части и дробной части числа:</p> <p>Целая часть: <math>11011000_2</math>  Дробная часть: <math>11_2</math></p> <p>Таким образом, число -80 в формате с фиксированной точкой (в однобайтовом формате со знаком) в прямом коде будет иметь следующий вид:</p> $1\ 1011000\ 11$ <p>где первый бит - знак числа (1 - отрицательное число), следующие 6 бит - целая часть, оставшиеся 2 бит - дробная часть.</p>	
3	<p>Для рисунка размером 128 на 128 пикселей общее количество пикселей составляет <math>128 \times 128 = 16384</math>.</p> <p>Обычно цвет каждого пикселя кодируется 24 битами - по 8 бит на каждый из каналов RGB. Таким образом, размер одного пикселя составляет <math>24 / 8 = 3</math> байта.</p> <p>Тогда размер изображения в байтах равен <math>16384 \times 3 = 49152</math> байта.</p> <p>Чтобы найти максимальное количество цветов в палитре, нужно разделить объем памяти, выделенный для палитры, на размер одного цвета в байтах. Пусть размер палитры равен n, тогда:</p> $n \times \text{размер одного цвета в байтах} = 250 \text{ Кбайт} = 250 \times 1024 \text{ байт.}$ <p>Размер одного цвета зависит от количества бит, выделенных для кодирования цвета в палитре. Обычно используются 8 бит (256 возможных значений) или 24 бита (16.7 миллионов возможных значений). Поскольку мы ищем максимальное количество цветов, то будем использовать 24-битную палитру.</p> <p>Тогда размер одного цвета в байтах равен <math>24 / 8 = 3</math> байта.</p> <p>Итак, получаем:</p> $n \times 3 = 250 \times 1024 \text{ байт.}$ <p>Отсюда:</p> $n = (250 \times 1024) / 3 = 85333.33 \text{ (округляем до ближайшего целого числа)}$	5

	<p>Таким образом, максимальное количество цветов в палитре изображения равно 85333.+</p>	
4	<p>Для расчета размера файла нам нужно знать сколько байтов занимает каждый сэмпл звуковой дорожки и сколько всего сэмплов будет записано в течение 1.5 минут.</p> <p>Четырехканальная звукозапись с 16-битным разрешением значит, что каждый сэмпл будет занимать <math>4 * 2 = 8</math> байтов (так как для каждого из 4 каналов используется 2 байта). Частота дискретизации 16 кГц значит, что в течение одной секунды будет записано 16000 сэмплов. Значит, за 1.5 минуты (90 секунд) будет записано <math>90 * 16000 = 1440000</math> сэмплов.</p> <p>Таким образом, размер файла можно рассчитать, умножив количество сэмплов на размер каждого сэмпла в байтах и разделив на 1 мегабайт (1048576 байтов):</p> <p>размер_файла = <math>(1440000 \text{ сэмплов}) * (8 \text{ байтов/сэмпл}) / (1048576 \text{ байтов/мегабайт}) = 10.875</math> мегабайт (округляем до 2 знаков после запятой).</p> <p>Ответ: размер полученного файла составит примерно 10.88 МБ.</p>	5
5	 <pre> graph TD     Start([начало]) --&gt; Input[/ввод n/]     Input --&gt; Init[count = 0]     Init --&gt; Loop{i = 0; i &lt; n; i++}     Loop --&gt; Input2[/ввод num/]     Input2 --&gt; Decision{num &gt; 8}     Decision -- да --&gt; Increment[count++]     Decision -- нет --&gt; Loop     Increment --&gt; Loop     Loop --&gt; Output[/count/]   </pre>	10

6	<pre># Запрашиваем у пользователя значение n n = int(input("Введите значение n: ")) # Инициализируем счетчик и сумму fact = 1 sum = 0 # Проходим по каждому числу от 1 до n for i in range(1, n+1):     # Вычисляем факториал числа i     fact *= i     # Добавляем факториал числа i к сумме     sum += fact # Выводим результат print(f"Сумма факториалов от 1 до {n} равна: {sum}")</pre>	10
7	<pre># Задаем массив K K = [3, 5, 1, 4, 6, 7, 2, 9, 8] # Инициализируем переменные для счетчика нечетных элементов и их суммы count = 0 sum_odd = 0 # Проходим по каждому элементу массива for i in K:     # Если элемент нечетный, добавляем его к сумме и увеличиваем счетчик     if i % 2 != 0:         sum_odd += i         count += 1 # Вычисляем среднее арифметическое нечетных элементов if count &gt; 0:     avg_odd = sum_odd / count else:     avg_odd = 0 # Выводим результат print("Среднее арифметическое нечетных элементов в массиве K:", avg_odd)</pre>	10

8	<pre> matrix = [[1, 2, 3], [4, 5, 6], [7, 8, 9]] for row in matrix:     sum_odd = 0     count_odd = 0     for i in range(len(row)):         if i % 2 != 0:             sum_odd += row[i]             count_odd += 1     if count_odd &gt; 0:         avg_odd = sum_odd / count_odd         print(f"Среднее арифметическое нечетных элементов строки {row}: {avg_odd}")     else:         print(f"В строке {row} нет нечетных элементов") </pre>	15
9	<pre> n = int(input("Введите натуральное число: ")) while n &gt;= 5:     n /= 5 if n == 1:     print("Число является степенью числа 5") else:     print("Число не является степенью числа 5") </pre>	15
10	<pre> N = [2, 5, 7, 3, 9, 8, 4] # пример входного массива P = 6 # пример заданного числа sum_n = 0 # инициализируем сумму чисел, меньших P count_n = 0 # инициализируем количество таких чисел # проходимся по каждому элементу массива N for n in N:     if n &lt; P: # если число меньше P         sum_n += n # добавляем его к сумме         count_n += 1 # увеличиваем счетчик на 1  if count_n &gt; 0: # если есть хотя бы одно число меньше P     avg_n = sum_n / count_n # считаем среднее арифметическое     print("Среднее арифметическое чисел, меньших", P, "равно", avg_n) else: # если нет чисел меньше P     print("В массиве нет чисел, меньших", P) </pre>	20



**Оборонно-техническая олимпиада 2022-2023 гг. (II этап)**

**Типовое задание для 11 класса**

**Направление: Информатика и программирование**

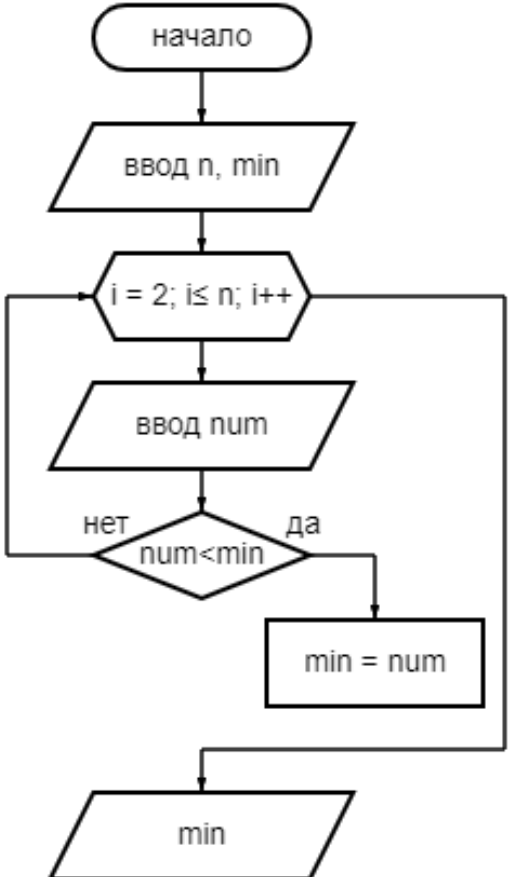
1	Перевести в двоичную систему счисления числа: 12410; 158; AAF16.
2	Сколько значащих нулей в двоичной записи шестнадцатеричного числа 50AC16
3	Каков информационный объем текстового файла в Кбайтах, содержащего 2 страницы текста при кодировке Unicode (алфавит в 65536 символов), если каждая страница содержит в среднем 25 строк по 50 символов в каждой?
4	Сканируется цветное изображение формата А4 (21*29,7 см). Разрешающая способность сканера – 1000*1000 dpi, он может различать 16777216 (=24) цветов. Какой информационный объем будет иметь полученный графический файл?
5	Составить и нарисовать блок схему в соответствии с задачей С клавиатуры вводится последовательность из N чисел, найти минимальное из них.
6	Дана матрица P. Найти сумму элементов предпоследней строки и заменить последний элемент этой строки полученной суммой.
7	В заданной матрице определить сумму элементов, больших среднего арифметического матрицы и меньших заданному числу J.
8	Дана матрица целых чисел. В каждой строке найти среднее арифметическое элементов с нечетными положительными индексами.
9	В заданной матрице определить сумму элементов, меньших среднего арифметического матрицы.
10	В каждой строке заданной матрицы определить количество нечетных элементов и сформировать из них одномерный массив.

## Решение

### 11 класс

1	<p>Переводим целую часть 12410 в 2-ую систему последовательным делением на 2:</p> $124/2 = 62, \text{ остаток: } 0$ $62/2 = 31, \text{ остаток: } 0$ $31/2 = 15, \text{ остаток: } 1$ $15/2 = 7, \text{ остаток: } 1$ $7/2 = 3, \text{ остаток: } 1$ $3/2 = 1, \text{ остаток: } 1$ $1/2 = 0, \text{ остаток: } 1$ $12410 = 11111002$ <p>Переводим 158 в десятичную систему:</p> $158 = 1 \cdot 8 \cdot 1 + 5 \cdot 1 = 1310$ <p>Переводим целую часть 1310 в 2-ую систему последовательным делением на 2:</p> $13/2 = 6, \text{ остаток: } 1$ $6/2 = 3, \text{ остаток: } 0$ $3/2 = 1, \text{ остаток: } 1$ $1/2 = 0, \text{ остаток: } 1$ $1310 = 11012$ <p>Переводим AAF16 в десятичную систему:</p> $AAF16 = 10 \cdot 16^2 + 10 \cdot 16 \cdot 1 + 15 \cdot 1 = 273510$ <p>Переводим целую часть 273510 в 2-ую систему последовательным делением на 2:</p> $2735/2 = 1367, \text{ остаток: } 1$ $1367/2 = 683, \text{ остаток: } 1$ $683/2 = 341, \text{ остаток: } 1$ $341/2 = 170, \text{ остаток: } 1$ $170/2 = 85, \text{ остаток: } 0$ $85/2 = 42, \text{ остаток: } 1$ $42/2 = 21, \text{ остаток: } 0$ $21/2 = 10, \text{ остаток: } 1$ $10/2 = 5, \text{ остаток: } 0$ $5/2 = 2, \text{ остаток: } 1$ $2/2 = 1, \text{ остаток: } 0$ $1/2 = 0, \text{ остаток: } 1$ $273510 = 1010101011112$	5
---	--	---

2	<p>Переведем шестнадцатеричное число 50AC в двоичную систему счисления: 5 = 0101, 0 = 0000, A = 1010, C = 1100 Таким образом, получаем число 0101000010101100 в двоичной системе счисления. Количество значащих нулей равно 6.</p>	5
3	<p>Количество символов на одной странице: <math>25 * 50 = 1250</math> символов.  Общее количество символов в двух страницах: <math>2 * 1250 = 2500</math> символов.  Каждый символ в кодировке Unicode занимает 2 байта (или 16 бит).  Общий объем файла в байтах: <math>2500 * 2 = 5000</math> байт.  Чтобы перевести байты в килобайты, нужно разделить на 1024:  <math>5000 / 1024 = 4,88</math> Кбайт (округляем до двух знаков после запятой).  Таким образом, информационный объем текстового файла составляет приблизительно 4,88 Кбайт.</p>	5
4	<p>Для решения задачи необходимо определить, сколько бит информации содержится в каждом пикселе изображения, а затем умножить эту величину на общее количество пикселей в изображении.  Размер листа А4 в дюймах можно найти, умножив соответствующие размеры в сантиметрах на коэффициент перевода 0,3937:  <math>21 \text{ см} * 0,3937 = 8,27</math> дюймов  <math>29,7 \text{ см} * 0,3937 = 11,69</math> дюймов  Таким образом, размер изображения в пикселях будет составлять:  <math>8,27 \text{ дюймов} * 1000 \text{ точек/дюйм} = 8270</math> точек по ширине  <math>11,69 \text{ дюймов} * 1000 \text{ точек/дюйм} = 11690</math> точек по высоте  Общее количество пикселей в изображении:  <math>8270 * 11690 = 96662300</math> точек  Каждый пиксель содержит информацию о цвете, которую можно представить в виде 24-битного числа (т.е. 3 байта или <math>3*8=24</math> бит).  Следовательно, общий объем информации в файле будет равен:  <math>96662300 * 24 \text{ бит} = 2327903200</math> бит  Переведем полученный результат в мегабайты, разделив на 8 и на 1 000 000:  <math>2327903200 / 8 / 1000000 = 291,99</math> Мб  Ответ: при сканировании цветного изображения формата А4 разрешением <math>1000*1000</math> dpi и различающейся палитре в 16777216 цветов, информационный объем полученного графического файла составит около 291,99 Мб.</p>	5

5	 <pre> graph TD     Start([начало]) --&gt; Input[/ввод n, min/]     Input --&gt; LoopStart{i = 2; i ≤ n; i++}     LoopStart --&gt; InputNum[/ввод num/]     InputNum --&gt; Decision{num &lt; min}     Decision -- да --&gt; AssignMin[min = num]     AssignMin --&gt; LoopStart     Decision -- нет --&gt; LoopStart     LoopStart --&gt; Output[/min/]   </pre>	10
6	<pre> import random # создаем матрицу matrix = [[random.randint(1, 10) for j in range(5)] for i in range(5)] print("Матрица:") for row in matrix:     print(row) # вычисляем среднее арифметическое элементов матрицы sum_elements = 0 count_elements = 0 for row in matrix:     for element in row:         sum_elements += element         count_elements += 1 average = sum_elements / count_elements print("Среднее арифметическое элементов матрицы:", average) # вычисляем сумму элементов, меньших среднего арифметического sum_less_average = 0 for row in matrix:     for element in row: </pre>	10

	<pre> if element &lt; average:     sum_less_average += element print("Сумма элементов, меньших среднего арифметического:", sum_less_average) </pre>	
7	<pre> # получаем матрицу от пользователя matrix = [] n = int(input("Введите количество строк: ")) m = int(input("Введите количество столбцов: ")) for i in range(n):     row = []     for j in range(m):         row.append(int(input(f"Введите элемент [{i}][{j}]: ")))     matrix.append(row) # находим количество нечетных элементов в каждой строке odd_counts = [] for row in matrix:     odd_count = 0     for element in row:         if element % 2 != 0:             odd_count += 1     odd_counts.append(odd_count)  # формируем одномерный массив из количества нечетных элементов odd_array = [] for count in odd_counts:     odd_row = []     for i in range(count):         odd_row.append(1)     odd_array.append(odd_row) print("Матрица:") for row in matrix:     print(row) print("Количество нечетных элементов в каждой строке:", odd_counts) print("Массив из нечетных элементов в каждой строке:", odd_array) </pre>	10
8	<pre> # Задаем матрицу P P = [[1, 2, 3], [4, 5, 6], [7, 8, 9], [10, 11, 12]] </pre>	15

	<pre> # Находим индекс предпоследней строки row_index = len(P) - 2  # Вычисляем сумму элементов предпоследней строки sum_row = sum(P[row_index])  # Заменяем последний элемент этой строки полученной суммой P[row_index][-1] = sum_row # Выводим измененную матрицу P print("Измененная матрица P:") for row in P:     print(row) </pre>	
9	<pre> # Задаем матрицу A и число J A = [[1, 2, 3], [4, 5, 6], [7, 8, 9]] J = 6 # Вычисляем среднее арифметическое матрицы sum_matrix = 0 count = 0 for row in A:     for elem in row:         sum_matrix += elem         count += 1 if count &gt; 0:     avg_matrix = sum_matrix / count else:     avg_matrix = 0 # Вычисляем сумму элементов, больших среднего арифметического матрицы и меньших J sum_elements = 0 for row in A:     for elem in row:         if elem &gt; avg_matrix and elem &lt; J:             sum_elements += elem # Выводим результат print("Сумма элементов матрицы, больших среднего арифметического матрицы и меньших числа J:", sum_elements) </pre>	15

10	<pre># Задаем матрицу A A = [[1, 2, 3], [4, 5, 6], [7, 8, 9]] # Вычисляем среднее арифметическое элементов с нечетными положительными индексами в каждой строке for row in A:     sum_odd_elems = 0     count_odd_elems = 0     for i in range(1, len(row), 2):         if row[i] &gt; 0:             sum_odd_elems += row[i]             count_odd_elems += 1     if count_odd_elems &gt; 0:         avg_odd_elems = sum_odd_elems / count_odd_elems     else:         avg_odd_elems = 0     print("Среднее арифметическое нечетных положительных элементов в строке", row, "равно", avg_odd_elems)</pre>	20
----	--	----