

Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение
высшего образования

**«ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
СИСТЕМ УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ» (ТУСУР)**

ИНФОРМАТИКА

Методические указания для подготовки к вступительным экзаменам,
проводимым ТУСУРом самостоятельно по дисциплине «ИНФОРМАТИКА»

Потахова Ирина Владимировна

Методические указания для подготовки к вступительным экзаменам, проводимым ТУ-СУРОм самостоятельно по дисциплине «ИНФОРМАТИКА» / И.В. Потахова. – Томск, 2020. – 60 с.

© Томский государственный университет систем
управления и радиоэлектроники, 2020
© Потахова И.В., 2020

Оглавление

1 Введение	4
2 Пример билета работы по информатике и его решение	5
2.1 Вариант заданий	5
2.2 Решение и оформление ответов варианта заданий	8
3 Критерии оценивания работы по информатике	14
4 Материал для самостоятельной подготовки	15
4.1 Системы счисления	15
4.1.1 Общие понятия и примеры решения задач	15
4.1.2 Задания для самостоятельной подготовки	20
4.2 Информация и ее кодирование	21
4.2.1 Общие понятия и примеры решения задач	21
4.2.2 Задания для самостоятельной подготовки	27
4.3 Основы логики	30
4.3.1 Общие понятия и примеры решения задач	30
4.3.2 Задания для самостоятельной подготовки	34
4.4 Элементы теории алгоритмов. Задания для самостоятельной подготовки	37
4.5 Программирование. Задания для самостоятельной подготовки	53
5 Ответы	57
5.1 Системы счисления	57
5.2 Информация и ее кодирование	58
5.3 Основы логики	58
5.4 Элементы теории алгоритмов	58
6 Список рекомендуемой литературы	60

1 Введение

Успешность выполнения работы по информатике определяется подготовленностью выпускника по предмету. Предлагаемый материал предназначен для самостоятельной работы учащихся выпускных классов, и определяет круг знаний, который требуется для успешной сдачи вступительных испытаний в ТУСУР.

Важно знать структуру работы по информатике. Работа состоит из 12 заданий. Задания охватывают следующие тематические разделы курса:

1. системы счисления;
2. информация и ее кодирование;
3. основы логики;
4. элементы теории алгоритмов;
5. программирование.

Для успешного выполнения заданий первого раздела необходимо изучить следующие вопросы: понятие позиционных систем счисления, правила перевода чисел из одной системы счисления в другую (рассматриваются двоичная, троичная, пятеричная, семеричная, восьмеричная, десятичная, шестнадцатеричная системы счисления), сложение и вычитание в позиционных системах счисления, умножение, деление в двоичной системе счисления.

При выполнении заданий второго раздела проверяются знания о методах измерения количества информации, умение подсчитывать информационного объема сообщения, умение кодировать и декодировать информацию.

Задания третьего раздела требуют знания логических законов и правил преобразование логических выражений.

При выполнении заданий четвертого раздела проверяются навыки формального исполнения алгоритма, умения анализировать программу, содержащую циклы и ветвления, программу, использующую процедуры и функции и функции.

В разделе «Программирование» проверяется умение создавать собственные программы и записывать их на одном из алгоритмических языков программирования общего назначения. Примеры возможных задач:

1. выделение цифр целого числа и их анализ;
2. преобразование целого числа;
3. разложение целого числа на множители;
4. вычисление среднего арифметического чисел;
5. поиск минимального или максимального значения;
6. суммирование элементов массива;
7. вычисление произведения элементов массива;
8. проверка упорядоченности массива и т.п.

Данные методические указания содержат следующее:

- пример билета работы по информатике с указанием максимального балла, который соответствуют правильному решению заданий билета;
- пример решения заданий билета и оформления ответов;
- критерии оценивания выполнения заданий билета по информатике;
- примеры решения типовых задач из разделов:
 1. системы счисления;
 2. информация и ее кодирование;
 3. основы логики;
- задачи для самостоятельной подготовке по всем темам, включенным в билет по информатике.

2 Пример билета работы по информатике и его решение

2.1 Вариант заданий

1. Вычислите выражение $2110_3 - 48_9 + 5_{10}$. Ответ дайте в десятичной системе счисления. **(6 баллов)**

2. Михаил составляет 5-буквенные коды. В кодах разрешается использовать только буквы А, Б, В, Г, Д, при этом код не может начинаться с гласной и не может содержать двух одинаковых букв подряд. Сколько различных кодов может составить Михаил? **(7 баллов)**

3. Автомат обрабатывает натуральное число $N > 1$ по следующему алгоритму.

1. Строится двоичная запись числа N .

2. Последняя цифра записи удаляется.

3. Если исходное число N было нечетным, в конец записи (справа) дописываются цифры 10, если четным – 01.

4. Результат переводится в десятичную систему счисления и выводится на экран. Какое число нужно ввести в автомат, чтобы в результате получилось 2017? **(7 баллов)**

4. Для какого наименьшего целого неотрицательного числа A выражение $(2x + 3y > 30) \vee (x + y \leq A)$ тождественно истинно, т.е. принимает значение 1 при любых целых неотрицательных x и y ? **(7 баллов)**

5. Сколько различных решений имеет уравнение $(K \wedge L) \vee (M \wedge N) = 1$, где K, L, M, N – логические переменные?

В ответе не нужно перечислять все различные наборы значений K, L, M и N , при которых выполнено данное равенство. В качестве ответа вам нужно указать только количество таких наборов. **(8 баллов)**

6. Определите, что будет напечатано в результате работы следующего фрагмента программы: **(8 баллов)**

Бейсик	Python	Паскаль	Си++
DIM K, S AS INTEGER S = 0 K = 0 WHILE S < 80 S = S + 2*K K = K + 4 WEND PRINT S	s = 0 k = 0 while s < 80: s += 2*k k += 4 print(s)	var k, s: integer; begin s:=0; k:=0; while s < 80 do begin s:=s+2*k; k:=k+4; end; write(s); end.	#include <iostream> using namespace std; int main() { int s, k; s = 0, k = 0; while (s < 80) { s = s + 2*k; k = k + 4; } cout << s << endl; return 0; }

7. Чему будет равно значение, вычисленное рекурсивным алгоритмом при выполнении вызова F(5)? **(9 баллов)**

Си	Python
<pre>int F(int n) { if (n > 2) return F(n-1)+F(n-2)+F(n-3); else return n; }</pre>	<pre>def F(n): if n > 2: return F(n-1)+F(n-2)+F(n-3) else: return n</pre>
Бейсик	Паскаль
<pre>FUNCTION F(n) IF n > 2 THEN F = F(n-1)+F(n-2)+F(n-3) ELSE F = n END IF END FUNCTION</pre>	<pre>function F(n: integer): integer; begin if n > 2 then F := F(n-1)+F(n-2)+F(n-3) else F := n; end;</pre>

8. В программе используется одномерный целочисленный массив A с индексами от 0 до 9. Значения элементов равны 5; 1; 6; 7; 8; 8; 7; 7; 6; 9 соответственно, т.е. A[0] = 5; A[1] = 1 и т.д. Определите значение переменной c после выполнения следующего фрагмента программы, записанного ниже на разных языках программирования. **(9 баллов)**

Бейсик	Python	Паскаль	Си++
<pre>c = 0 FOR i = 1 TO 9 IF A(i - 1) >= A(i) THEN t = A(i) A(i) = A(i - 1) A(i - 1) = t ELSE c = c + 1 ENDIF NEXT i</pre>	<pre>c = 0 for i in range(1, 10): if A[i - 1] >= A[i]: t = A[i] A[i] = A[i - 1] A[i - 1] = t else: c = c + 1</pre>	<pre>c := 0; for i := 1 to 9 do if A[i - 1] >= A[i] then begin t := A[i]; A[i] := A[i - 1]; A[i - 1] := t end else c := c + 1;</pre>	<pre>c = 0; for (i = 1; i <= 9; i++) if (A[i-1] >= A[i]) { t = A[i]; A[i] = A[i - 1]; A[i - 1] = t; } else c++;</pre>

9. Напишите в ответе наименьшее значение входной переменной k, при котором программа выдаёт ответ 21. Для Вашего удобства программа приведена на четырех языках программирования. **(9 баллов)**

Бейсик	Python
<pre>DIM K, I AS LONG INPUT K I = 1</pre>	<pre>def f(n): return n*n*n</pre>

<pre> WHILE F(I) <= K*G(I) I = I + 1 WEND PRINT I FUNCTION F(N) F = N * N * N END FUNCTION FUNCTION G(N) G = N * N END FUNCTION </pre>	<pre> def g(n): return n*n k = int(input()) i = 1 while f(i) <= k*g(i): i += 1 print (i) </pre>
Паскаль	Си++
<pre> var k, i : longint; function f(n: longint): longint; begin f := n * n * n; end; function g(n: longint): longint; begin g := n*n; end; begin readln(k); i := 1; while f(i) <= k*g(i) do i := i+1; writeln(i) end. </pre>	<pre> #include <iostream> using namespace std; long f(long n) { return n * n * n; } long g(long n) { return n*n; } int main() { long k, i; cin >> k; i = 1; while (f(i) <= k*g(i)) i++; cout << i << endl; return 0; } </pre>

10. Дано целое число N (> 2). Написать программу, которая проверяет, является ли введенное число N простым. Число N является простым, если его единственными делителями являются 1 и само число N . **(10 баллов)**

11. Дан одномерный массив размера N (N — четное число). Элементы массива — целые числа. Найти номера двух ближайших элементов из этого массива (то есть элементов с наименьшим модулем разности значений) и вывести эти номера в порядке возрастания. Вспомогательный массив не использовать. **(10 баллов)**

12. Специальная камера, установленная на перекрёстке, фиксирует количество проезжающих автомобилей, и каждую минуту по каналу связи передаёт неотрицательное целое число — количество автомобилей, проехавших перекрёсток за эту минуту. Известно, что за минуту перекрёсток может проехать не более 100 автомобилей. Напишите на любом языке программирования эффективную, в том числе и по памяти,

программу, которая находит в заданной серии показаний максимальное количество автомобилей, проехавших перекрёсток в течение пяти подряд идущих минут. Максимальное количество показаний, которое может передать камера, не превышает 1440.

Входные данные представлены следующим образом. В первой строке подаётся число N – общее количество переданных показаний. Гарантируется, что $N > 5$. В следующей строке через пробел записано N натуральных чисел – очередное показание камеры.

Пример входных данных: 8

5 12 27 10 4 50 7 16

Пример выходных данных для приведённого выше примера входных данных: 103
(10 баллов)

2.2 Решение и оформление ответов варианта заданий

1. Решение

Представим все числа выражения $2110_3 - 48_9 + 5_{10}$ в десятичной системе счисления.

$$2110_3 = 2 \cdot 3^3 + 1 \cdot 3^2 + 1 \cdot 3^1 + 0 \cdot 3^0 = 66_{10}$$

$$48_9 = 4 \cdot 9^1 + 8 \cdot 9^0 = 44_{10}$$

$$66 - 44 + 5 = 27$$

Ответ: 27

2. Решение

На первое место слова можно поставить любую из 5 букв, кроме буквы А. На каждое следующее место можно поставить любую из 5 букв, кроме той, которая стоит на предыдущем месте. Значит, всего Михаил может составить $4 \cdot 4 \cdot 4 \cdot 4 \cdot 4 = 1024$ кода.

Ответ: 1024

3. Решение

Рассмотрим действие алгоритма в обратном порядке. Переведем число 2017 в двоичную систему счисления:

$2017_{10} = 1111110000_2$. Последние две двоичные цифры – 01, следовательно, исходное число было четным. Рассмотрим числа, из которых можно получить натуральное число N .

$$1111110000_2 = 1008_{10}$$

$$1111110001_2 = 1009_{10}$$

Ответ: 1008

4. Решение.

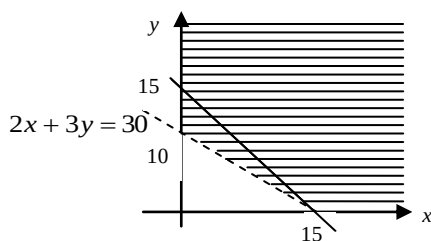
Разделим исходное выражение $(2x + 3y > 30) \vee (x + y \leq A)$ на две части:

- известная часть $2x + 3y > 30$;
- неизвестная часть $x + y \leq A$.

Определим условия истинности исходного выражения.

- Если а) истинно, а б) ложно.
- Если а) ложно, а б) истинно.
- Если истинны одновременно а) и б).

Если для некоторых целых положительных x и y ($2x + 3y > 30$), то исходное выражение истинно независимо от значения A (на рисунке выделено заштрихованной областью).



Поэтому будем рассматривать только область, в которой $2x + 3y \leq 30$ (белая область, включая точки прямой $2x + 3y = 30$).

Необходимо найти такое A , чтобы для всех x и y из этой области выполнялось условие $x + y \leq A$. Очевидно, A должно быть равно наибольшему возможному значению $x + y$ в этой области. На рисунке видно, что прямая $x + y = A$ должна быть правой границей области, определяемой условием $x + y \leq A$ и должна проходить через точку $(15, 0)$. Таким образом, наименьшее целое неотрицательное A равно 15.

Ответ: 15

5. Решение.

Выражение $(K \wedge L) \vee (M \wedge N) = 1$ истинно в трех случаях, когда $(K \wedge L)$ и $(M \wedge N)$ равны соответственно 01, 10, 11.

Составим таблицу решений.

1. "01":	$K \wedge L = 0;$		$M \wedge N = 1.$	
	K	L	M	N
1	1	0	1	1
2	0	1	1	1
<u>3</u>	0	0	1	1
2. "10":	$K \wedge L = 1;$		$M \wedge N = 0.$	
	K	L	M	N
1	1	1	1	0
2	1	1	0	1
<u>3</u>	1	1	0	0
3. "11":	$K \wedge L = 1;$		$M \wedge N = 1$	
	K	L	M	N
<u>1</u>	1	1	1	1

Ответ: 7

6. Решение.

Выполним указанный фрагмент программы, записывая в таблицу результаты

Шаг	1	2	3	4	5
Переменная					
s	0	8	24	48	80
k	4	8	12	16	20

Ответ: 80

7. Решение

1. Вычисленное алгоритмом значение F равно.

$$F(5) = F(4) + F(3) + F(2) = F(3) + F(2) + F(1) + F(2) + F(1) + F(0) + 2 = F(2) + F(1) + F(0) + F(2) + F(1) + F(2) + F(1) + F(0) + 2 = 2 + 1 + 0 + 2 + 1 + 2 + 1 + 0 + 2 = 11$$

Ответ: 11

либо

2. Запишем все значения функции F:

$$F(0) = 0$$

$$F(1) = 1$$

$$F(2) = 2$$

$$F(3) = F(2) + F(1) + F(0) = 3$$

$$F(4) = F(3) + F(2) + F(1) = 6$$

$$F(5) = F(4) + F(3) + F(2) = 11$$

Ответ: 11

8. Решение.

Значение переменной s увеличивается на единицу, если значение элемента, стоящего на текущей позиции, меньше значения элемента, стоящего на предыдущей позиции. В противном случае соответствующие элементы меняются местами.

Элементы массива	5	1	6	7	8	8	7	7	6	9
	↔									
	1 5									
	5	6								
		6	7							
			7	8						
				↔						
				8 8						
					↔					
					7 8					
						↔				
						7 8				
							↔			
							6 8			
								8	9	
Шаг (i)		1	2	3	4	5	6	7	8	9

Для заданного массива выполнено 5 перестановок, следовательно, значение переменной $s = 9 - 5 = 4$.

Ответ: 4

9. Решение:

Цикл *While* завершится, когда выполнится неравенство $i^3 > k \cdot i^2$.

Задано условие окончания цикла: $21^3 > k \cdot 21^2$. Следовательно, последний шаг цикла определяется неравенством $20^3 \leq k \cdot 20^2$. Находим $k = 20$.

Ответ: $k = 20$.

10. Решение:

Для решения этой задачи реализуем следующий алгоритм:

Опишем целочисленную переменную N (заданное число) и запросим ее значение с клавиатуры.

Опишем целочисленную переменную i , которую в дальнейшем будем использовать как переменную – параметр цикла и очередной делитель числа.

Опишем целочисленную переменную k – остаток от целочисленного деления заданного числа N на делитель i .

Инициализируем начальное значение переменной $i = 2$ (первый делитель).

Инициализируем начальное значение переменной $k = 1$ (число простое).

Организуем цикл вычисления очередного остатка k .

Цикл будет продолжаться пока $k \geq 1$ (i – не является делителем N) и $i \leq N/2$ (не перебраны все делители числа N).

После цикла значение переменной $k \neq 0$, если заданное число N простое.

Решение на языке Си представлено ниже:

```
#include <iostream>
using namespace std;
int main()
{
    long N, i;
    cin >> N;
    i = 2; k = 1;
    while (k >= 1 && i <= N/2)
    {
        k = N % i;  i++;
    }
    if (k==0) cout << "Число составное" << endl; else cout << "Число простое" << endl;
    return 0;
}
```

11. Решение:

Для решения этой задачи реализуем следующий алгоритм. Опишем целочисленную переменную N для хранения количества элементов массива. Опишем статический массив X из 100 элементов. Организуем ввод значения размерности массива с клавиатуры. Проверим, что введенное с клавиатуры значение не превышает размера статического массива. Для организации ввода элементов массива с клавиатуры используем цикл, который будет выполняться N раз.

Для нахождения элементов массива, модуль разности которых минимален необходимо найти модули разности значений всех возможных пар элементов и выбрать из них минимальный. Для сохранения номеров указанных элементов опишем в программе переменные $i1$ и $i2$. Будем считать ближайшими элементами нулевой и первый элементы массива, то есть определим $i1 = 0$, $i2 = 1$, в переменной min сохраним модуль

разности значений элементов $X[0]$ и $X[1]$. Для перебора всех возможных пар элементов организуем два вложенных цикла. Внешний цикл будет перебирать элементы от нулевого до $N-1$ (от первого до последнего) изменяя счетчик i , внутренний цикл будет перебирать элементы от $i+1$ -го (чтобы исключить обращение к ранее просмотренным парам элементов) до последнего элемента, изменяя счетчик j . Во внутреннем цикле организуем вычисления модуля разности значений элементов $X[i]$ и $X[j]$ и сравнение найденного модуля с текущим значением переменной **min**. Если найденный модуль разности меньше текущего минимального значения, перепределим значения переменных **min**, **i1** и **i2**.

После окончания циклов выведем значения переменных **i1** и **i2**. Дополнительной проверки для сравнения найденных индексов можно не предусматривать, поскольку из-за описанной выше комбинации циклов переменная **i1** всегда будет меньше переменной **i2**. Реализация алгоритма на языке Си приведена ниже.

```
#include <stdio.h>
#include <math.h>
#include <stdlib.h>
int main(int argc, char** argv) {
    int X[100], N, b, i, j, i1=0, i2=1;
    system("chcp 1251");
    printf("Введите размерность массива: ");
    scanf("%d", &N);
    if (N>100) {printf ("Ошибка.....");
        return 0;
    }
    printf("Вводите элементы массива: ");
    for (i=0; i<N; i++)
        scanf("%d", &X[i]);
    int min = abs(X[0]-X[1]);
    printf("\n");
    for (i=0; i<N; i++){
        for (j=i+1; j<N; j++){
            b = abs(X[i]-X[j]);
            if (b<min){min = b; i1 = i; i2 = j;
            }
        }
    }
    printf("Ближайшие элементы - %d и %d \n", i1, i2);
    return 0;
}
```

12. Решение:

Для решения этой задачи реализуем следующий алгоритм.

Опишем в программе переменную N для хранения количества входных данных. Для оптимизации программы по памяти будем хранить только серию из пяти текущих значений, для этого опишем в программе массив для хранения только пяти переменных – $X[5]$. Организуем цикл для ввода пяти первых значений в массив X . За максимальное количество машин, примем количество машин, проехавших перекресток за первые пять минут, для этого опишем переменную **max** и сохраним в ней сумму элементов массива. Это же значение сохраним в переменной **sum**. В дальнейшем пере-

менную **sum**, будем использовать для хранения количества машин, проехавших перекресток в течение текущих пяти минут.

Для организации ввода оставшихся данных используем цикл, изменяющий счетчик **i**, который будет повторяться **N-5** раз. Опишем переменную **a**, в которую будем вводить значение количества машин, проехавших перекресток в **i-тую** минуту. В цикле будем выполнять следующие действия:

- ввод количества машин;
- уменьшение переменной **sum** на значение **X[0]** и увеличение на значение **a** (таким образом организовано вычисление количества машин, проехавших перекресток за текущие 5 минут);
- сдвиг элементов в массиве **X** на одну позицию влево и заполнение последнего элемента массива считанным значением **a**;
- сравнение значения **sum** с текущим значением максимальной суммы и изменение значения переменной **max**, если значение **sum > max**.

После выполнения суммы организуем вывод переменной **max**.

Ниже представлен код программы на языке Си.

```
#include <stdio.h>
#include <stdlib.h>
int main(int argc, char** argv) {
    int X[5], max, sum = 0, i, a, N, j;
    system("chcp 1251");
    printf("Введите количество показаний: ");
    scanf("%d", &N);
    printf("Вводите показания: ");
    for (i=0; i<5; i++){
        scanf("%d", &X[i]);
        sum = sum + X[i];
    }
    max = sum;
    for (i=0; i<N-5; i++){
        scanf("%d", &a);
        sum = sum - X[0]+a;
        if (sum>max) max = sum;
        // сдвиг элементов массива данных влево на одну позицию
        for (j=0; j<4; j++)
            X[j]=X[j+1];
        X[j] = a;
    }
    printf("Максимальное количество машин, проехавших перекресток за 5 минут - %d \n", max) ;
    return 0;
}
```

3 Критерии оценивания работы по информатике

1. Работа оценивается в 100-бальной системе. Количество баллов за каждое задание указано в билете.

2. Задания 1-5 **не оцениваются**, если дан только ответ без соответствующего решения.

3. Задания 6-9 оценивается по максимуму, если дан ответ и выполнен анализ предложенного алгоритма. В противном случае задание оценивается от половины максимального балла.

4. Задания 10-12 оцениваются по максимуму, если соблюден синтаксис соответствующего языка программирования (допускается до трех синтаксических ошибок), описаны типы данных, используемые в программе, оформлены ввод исходных данных и вывод результата, выполнены требования соответствующего задания: однократный просмотр данных, не использование вспомогательного массива и т.п. В противном случае задания оцениваются по следующей схеме:

1. более 3 синтаксических ошибок – (максимальный балл – 2);
2. отсутствует или неверное описание данных – (максимальный балл – 3);
3. отсутствует ввод исходных данных или вывод результата – (максимальный балл – 4);
4. алгоритмические ошибки (отсутствие инициализации переменных, использование вспомогательного массива, неэффективное использование памяти и т.п.) – (максимальный балл – 5).

4 Задания для самостоятельной подготовки

4.1 Системы счисления.

4.1.1. Общие понятия и примеры решения задач

Для успешного выполнения заданий необходимо знать:

1. принципы кодирования чисел в позиционных системах счисления;
2. соответствие чисел, записанных в различных системах счисления (таблица 4.1);
3. правила перевода чисел между десятичной, двоичной, восьмеричной и шестнадцатеричной системами счисления;
4. сложение и вычитание в позиционных системах счисления;
5. умножение, деление в двоичной системе счисления.

Система счисления — это совокупность правил и приемов записи чисел с помощью набора цифровых знаков. Количество цифр, необходимых для записи числа в системе, называют основанием системы счисления. Основание системы записывается в справа числа в нижнем индексе: 5_{10} ; 1110110_2 ; $AF178_{16}$.

Таблица 4.1 Соответствие чисел, записанных в различных системах счисления

Десятичная	Двоичная	Восьмеричная	Шестнадцатеричная
1	001	1	1
2	010	2	2
3	011	3	3
4	100	4	4
5	101	5	5
6	110	6	6
7	111	7	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F
16	10000	20	10

1) Правила перевода чисел из одной системы счисления в другую

Перевод чисел из одной системы счисления в другую составляет важную часть машинной арифметики. Рассмотрим основные правила перевода.

1. Для перевода двоичного числа в десятичное необходимо его записать в виде многочлена, состоящего из произведений цифр числа и соответствующей степени числа 2, и вычислить по правилам десятичной арифметики:

$$X_2 = A_n \cdot 2^{n-1} + A_{n-1} \cdot 2^{n-2} + A_{n-2} \cdot 2^{n-3} + \dots + A_2 \cdot 2^1 + A_1 \cdot 2^0$$

При переводе удобно пользоваться таблицей степеней двойки (таблица 4.2):

Таблица 4.2 Степени числа 2

n	0	1	2	3	4	5	6	7	8	9	10
2^n	1	2	4	8	16	32	64	128	256	512	1024

Пример. Число 11101000_2 перевести в десятичную систему счисления.

$$11101000_2 = 1 \cdot 2^7 + 1 \cdot 2^6 + 1 \cdot 2^5 + 0 \cdot 2^4 + 1 \cdot 2^3 + 0 \cdot 2^2 + 0 \cdot 2^1 + 0 \cdot 2^0 = 232_{10}$$

2. Для перевода восьмеричного числа в десятичное необходимо его записать в виде многочлена, состоящего из произведений цифр числа и соответствующей степени числа 8, и вычислить по правилам десятичной арифметики:

$$X_8 = A_n \cdot 8^{n-1} + A_{n-1} \cdot 8^{n-2} + A_{n-2} \cdot 8^{n-3} + \dots + A_2 \cdot 8^1 + A_1 \cdot 8^0$$

При переводе удобно пользоваться таблицей степеней восьмерки (таблица 4.3):

Таблица 4.3 Степени числа 8

n	0	1	2	3	4	5	6
8^n	1	8	64	512	4096	32768	262144

Пример. Число 75013_8 перевести в десятичную систему счисления.

$$75013_8 = 7 \cdot 8^4 + 5 \cdot 8^3 + 0 \cdot 8^2 + 1 \cdot 8^1 + 3 \cdot 8^0 = 31243_{10}$$

3. Для перевода шестнадцатеричного числа в десятичное необходимо его записать в виде многочлена, состоящего из произведений цифр числа и соответствующей степени числа 16, и вычислить по правилам десятичной арифметики:

$$X_{16} = A_n \cdot 16^{n-1} + A_{n-1} \cdot 16^{n-2} + A_{n-2} \cdot 16^{n-3} + \dots + A_2 \cdot 16^1 + A_1 \cdot 16^0$$

При переводе удобно пользоваться таблицей степеней числа 16 (таблица 4.4):

Таблица 4.4 Степени числа 16

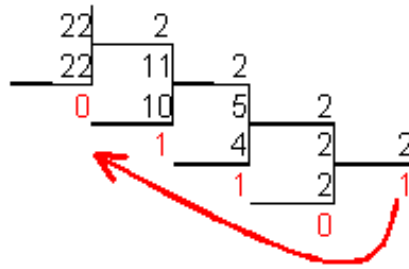
n	0	1	2	3	4	5	6
16^n	1	16	256	4096	65536	1048576	16777216

Пример. Число $FDA1_{16}$ перевести в десятичную систему счисления.

$$FDA1_{16} = 15 \cdot 16^3 + 13 \cdot 16^2 + 10 \cdot 16^1 + 1 \cdot 16^0 = 64929_{10}$$

4. Для перевода десятичного числа в двоичную систему его необходимо последовательно делить на 2 до тех пор, пока не останется остаток, меньший или равный 1. Число в двоичной системе записывается как последовательность последнего результата деления и остатков от деления в обратном порядке.

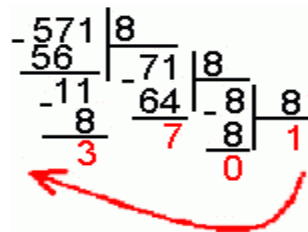
Пример. Число 22_{10} перевести в двоичную систему счисления.



$$22_{10} = 10110_2$$

5. Для перевода десятичного числа в восьмеричную систему его необходимо последовательно делить на 8 до тех пор, пока не останется остаток, меньший или равный 7. Число в восьмеричной системе записывается как последовательность цифр последнего результата деления и остатков от деления в обратном порядке.

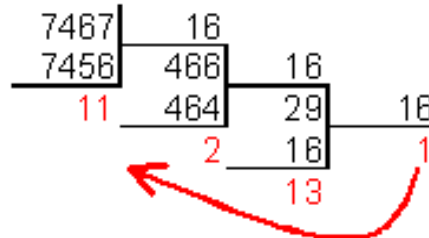
Пример. Число 571_{10} перевести в восьмеричную систему счисления.



$$571_{10} = 1073_8$$

6. Для перевода десятичного числа в шестнадцатеричную систему его необходимо последовательно делить на 16 до тех пор, пока не останется остаток, меньший или равный 15. Число в шестнадцатеричной системе записывается как последовательность цифр последнего результата деления и остатков от деления в обратном порядке.

Пример. Число 7467_{10} перевести в шестнадцатеричную систему счисления.



$$7467_{10} = 1D2B_{16}$$

7. Чтобы перевести число из двоичной системы в восьмеричную, его нужно разбить на триады (тройки цифр), начиная с младшего разряда, в случае необходимости дополнив старшую триаду нулями, и каждую триаду заменить соответствующей восьмеричной цифрой (табл. 3.1).

Пример. Число 1001011_2 перевести в восьмеричную систему счисления.

$$001\ 001\ 011_2 = 113_8$$

8. Чтобы перевести число из двоичной системы в шестнадцатеричную, его нужно разбить на тетрады (четверки цифр), начиная с младшего разряда, в случае необходимости дополнив старшую тетраду нулями, и каждую тетраду заменить соответствующей шестнадцатеричной цифрой (табл. 3.1).

Пример. Число 101110001_2 перевести в шестнадцатеричную систему счисления. $0011\ 1110\ 001_2 = 2E3_{16}$

9. Для перевода восьмеричного числа в двоичное необходимо каждую цифру заменить эквивалентной ей двоичной триадой.

Пример. Число 531_8 перевести в двоичную систему счисления.

$$531_8 = 101\ 011\ 001_2$$

10. Для перевода шестнадцатеричного числа в двоичное необходимо каждую цифру заменить эквивалентной ей двоичной тетрадой.

Пример. Число $EE8_{16}$ перевести в двоичную систему счисления.

$$EE8_{16} = 1110\ 1110\ 1000_2$$

11. При переходе из восьмеричной системы счисления в шестнадцатеричную и обратно, необходим промежуточный перевод чисел в двоичную систему.

Пример 1. Число FEA_{16} перевести в восьмеричную систему счисления.

$$FEA_{16} = 1111\ 1110\ 1010_2$$

$$111\ 111\ 101\ 010_2 = 7752_8$$

Пример 2. Число 6653_8 перевести в шестнадцатеричную систему счисления.

$$6653_8 = 110\ 110\ 101\ 011_2$$

$$1101\ 1010\ 1011_2 = DAB_{16}$$

2) Арифметические действия над целыми числами в 2-ой системе счисления.

1. Операция сложения выполняется с использованием таблицы двоичного сложения в одном разряде:

Пример.

а) $+ 1001_2$

$$\begin{array}{r} 1010_2 \\ + 1001_2 \\ \hline 10011_2 \end{array}$$

$$10011_2$$

б) $+ 1101_2$

$$\begin{array}{r} 1011_2 \\ + 1101_2 \\ \hline 11000_2 \end{array}$$

$$11000_2$$

в) $+ 11111_2$

$$\begin{array}{r} 1111_2 \\ + 11111_2 \\ \hline 100000_2 \end{array}$$

$$100000_2$$

$$\begin{array}{r|rr} + & 0 & 1 \\ \hline 0 & 0 & 1 \\ 1 & 1 & 10_2 \end{array}$$

2. Операция вычитания выполняется с использованием таблицы вычитания, в которой $\bar{1}$ обозначает заем в старшем разряде.

Пример.

а) -101110011_2

$$\begin{array}{r} 100011011_2 \\ - 101110011_2 \\ \hline 001011000_2 \end{array}$$

$$001011000_2$$

б) -110101101_2

$$\begin{array}{r} 101011111_2 \\ - 110101101_2 \\ \hline 001001110_2 \end{array}$$

$$001001110_2$$

$$\begin{array}{r|rr} - & 0 & 1 \\ \hline 0 & 0 & \bar{1}1_2 \\ 1 & 1 & 0 \end{array}$$

3. Операция умножения выполняется по обычной схеме, применяемой в десятичной с/с с последовательным умножением множимого на очередную цифру множителя.

$$\begin{array}{r|rr} \times & 0 & 1 \\ \hline 0 & 0 & 0 \\ 1 & 0 & 1 \end{array}$$

Пример.

$$\begin{array}{r}
 \text{а) } \times 11001_2 \\
 1101_2 \\
 11001 \\
 11001 \\
 11001 \\
 \hline
 101000101_2
 \end{array}
 \quad
 \begin{array}{r}
 \text{б) } \times 101_2 \\
 11_2 \\
 101 \\
 101 \\
 \hline
 1111_2
 \end{array}$$

4. Операция деления выполняется по алгоритму, подобному алгоритму выполнения операции деления в 10-ой системе счисления.

Пример.

$$\begin{array}{r|l}
 101000101_2 & 1101_2 \\
 1101 & 11001_2 \\
 \hline
 1110 & \\
 1101 & \\
 \hline
 1101 & \\
 1101 & \\
 \hline
 0 &
 \end{array}
 \quad
 \begin{array}{r|l}
 100001110_2 & 1111_2 \\
 1111 & 10010_2 \\
 \hline
 0001111 & \\
 1111 & \\
 \hline
 00 &
 \end{array}$$

3) Сложение и вычитание в восьмеричной системе счисления.

При выполнении сложения и вычитания в 8-ой с/с необходимо соблюдать следующие правила:

1) в записи результатов сложения и вычитания могут быть использованы только цифры восьмеричного алфавита;

2) десяток восьмеричной системы счисления равен 8, т.е. переполнение разряда наступает, когда результат сложения больше или равен 8.

В этом случае для записи результата надо вычесть 8, записать остаток, а к старшему разряду прибавить единицу переполнения;

3) если при вычитании приходится занимать единицу в старшем разряде, эта единица переносится в младший разряд в виде восьми единиц.

Пример

$$\begin{array}{r}
 + 770_8 \\
 236_8 \\
 \hline
 1226_8
 \end{array}
 \quad
 \begin{array}{r}
 - 750_8 \\
 236_8 \\
 \hline
 512_8
 \end{array}$$

4) Сложение и вычитание в шестнадцатеричной системе счисления.

При выполнении этих действий в 16-ой с/с необходимо соблюдать следующие правила:

1) при записи результатов сложения и вычитания надо использовать цифры шестнадцатеричного алфавита: цифры, обозначающие числа от 10 до 15 записываются латинскими буквами, поэтому, если результат является числом из этого промежутка, его надо записывать соответствующей латинской буквой;

2) десяток шестнадцатеричной системы счисления равен 16, т.е. переполнение разряда наступает, если результат сложения больше или равен 16, и в этом случае для

записи результата надо вычесть 16, записать остаток, а к старшему разряду прибавить единицу переполнения;

3) если приходится занимать единицу в старшем разряде, эта единица переносится в младший разряд в виде шестнадцати единиц.

Примеры.

$$\begin{array}{r} + \text{B09}_{16} \\ \text{EFA}_{16} \\ \hline 1\text{A03}_{16} \end{array} \quad \begin{array}{r} - \text{B09}_{16} \\ 7\text{FA}_{16} \\ \hline 30\text{F}_{16} \end{array}$$

4.1.2. Задания для самостоятельной подготовки

1. Переведите числа из 10-ой системы счисления в 2-ую систему счисления.

- 1) 165 2) 198 3) 541 4) 849
5) 127 6) 195 7) 289 8) 513.

2. Переведите числа из 2-ой в 10-ую систему счисления.

- 1) 110101 2) 100111 3) 1101100 4) 1011101
5) 11011101 6) 10010100 7) 111001010 8) 110001011.

3. Переведите числа из 2-ой в 8-ую, 16-ую системы счисления.

- 1) 100101110₂ 2) 100000111₂ 3) 111001011₂
4) 1000111011₂ 5) 1011001011₂ 6) 110011001011₂

4. Переведите числа из 10-ой в 8-ую, 16-ую системы счисления.

- 1) 69 2) 73 3) 113 4) 203 5) 351 6) 641

5. Переведите числа из 8-ой системы счисления в 10-ую систему счисления

- 1) 35₈ 2) 65₈ 3) 215₈ 4) 327₈ 5) 532₈ 6) 751₈

6. Какое число следует за каждым из данных:

- 1) 10₁₀; 2) AF₁₆;
3) 677₈; 4) 101₂.

Ответ для каждого числа запишите в указанной и десятичной системах счисления.

7. Какое число предшествует каждому из данных:

- 1) 10₁₀; 2) 9A₁₆;
3) 56₈; 4) 110₂.

8. Выполнить сложение в 2-ой системе счисления:

- 1) + 10010011₂ 2) + 1011101₂ 3) + 10110011₂
1011011₂ 11101101₂ 1010101₂

9. Выполнить вычитание в 2-ой системе счисления

- 1) - 100001000₂ 2) - 110101110₂ 3) - 11101110₂

- 10110011_2 10111111_2 1011011_2
10. Выполнить умножение в 2–ой системе счисления
- $1) \times 100001_2$ $2) \times 100101_2$ $3) \times 111101_2$
 111111_2 111011_2 111101_2
11. Выполнить деление в 2–ой системе счисления
- 1) $111010001001_2 / 111101_2$
 2) $100011011100_2 / 110110_2$
 3) $100001011110_2 / 111111_2$
12. Расставьте знаки арифметических операций так, чтобы были верны следующие равенства в двоичной системе:
- 1) $1100 ? 11 ? 100 = 100000$;
 2) $1100 ? 10 ? 10 = 100$;
 3) $1100 ? 10 ? 10 = 110000$;
 4) $1100 ? 10 ? 10 = 1011$;
 5) $1100 ? 11 ? 100 = 0$.
13. Вычислите выражения, записав результат в десятичной системе счисления:
- 1) $(1111101_2 + A_{F_{16}}) / 36_8$;
 2) $125_8 + 11101_2 \times A2_{16} - 1417_8$.
14. Решите уравнение $60_8 + x = 120_7$. Ответ запишите в шестеричной системе счисления. Основание системы счисления указывать не нужно.
15. Сумму восьмеричных чисел $17_8 + 1700_8 + 170000_8 + 17000000_8 + 1700000000_8$ перевели в шестнадцатеричную систему счисления. Найдите в записи числа, равного этой сумме, пятую цифру слева.
16. Восстановите неизвестные цифры, обозначенные знаком вопроса, в следующих примерах на сложение и вычитание, определив вначале, в какой системе изображены числа.
- 1)
$$\begin{array}{r} +5?55 \\ ?327 \\ \hline ?16?4 \end{array}$$
 2)
$$\begin{array}{r} -1536 \\ ?42 \\ \hline 67? \end{array}$$

4.2 Информация и ее кодирование

4.2.1. Общие понятия и примеры решения задачи

Для выполнения задания 2 по информатике необходимо уметь:

- 1) кодировать и декодировать информацию;
- 2) подсчитывать информационный объем сообщения.

1) Кодирование и декодирование информации

При решении задач этого типа необходимо знать понятия: кодирование и декодирование, код и длина кода.

Кодирование — это операция преобразования знаков или групп знаков одной знаковой системы в знаки или группы знаков другой знаковой системы.

Декодирование — это обратный процесс кодированию;

Кодирование бывает *равномерное* и *неравномерное*;

При **равномерном кодировании** все символы кодируются кодами равной длины.

Например:

Символ А кодируется последовательностью 00, символ Б – последовательностью 01, а символ В – последовательностью 10. В результате, например, сообщение "011000" может быть однозначно декодировано, как БВА. Здесь код каждого символа – это комбинация двух цифр 0 и 1, при этом длина кодов постоянна и равна 2.

При **неравномерном кодировании** разные символы могут кодироваться кодами разной длины, это затрудняет декодирование, связанные с появлением неоднозначности кода.

Например:

Символ А кодируется цифрой 0, Б – последовательностью 01, а В – последовательностью 1. В результате, например, сообщение "011" может быть декодировано, как АВВ или БВ. При неоднозначности кода информацию можно декодировать по разному. Для однозначного декодирования код должен удовлетворять **условию Фано**: никакое кодовое слово не может быть началом другого кодового слова.

Рассмотри примеры решения задач кодирования и декодирования информации.

Пример 1.

По каналу связи передаются сообщения, содержащие только семь букв: А, Б, Г, И, М, Р, Я. Для передачи используется двоичный код, удовлетворяющий условию Фано. Кодовые слова для некоторых букв известны: А — 010, Б — 00, Г — 101. Какое **наименьшее** количество двоичных знаков потребуется для кодирования слова МАГИЯ?

Примечание. Условие Фано означает, что ни одно кодовое слово не является началом другого кодового слова.

Решение.

По условию задачи определены коды трех букв: А – 010, Б – 00, Г – 101. Определим код для следующей буквы, причем по условию задания он должен иметь наименьшую длину. Рассмотрим коды 01, 10, 11.

Код 01 не подходит, поскольку код для буквы А не будет удовлетворять условию Фано. Поэтому для следующей буквы И определим код 011.

Код 10 не подходит, поскольку код для буквы Г не будет удовлетворять условию Фано. Поэтому для следующей буквы М определим код 100.

Следующую букву Р кодируем кодовым словом 110. Код 11 взять не можем, иначе не останется кодовых слов для оставшейся буквы Я, которые удовлетворяли бы условию Фано. Последняя буква будет кодироваться как 111.

Тогда наименьшее количество двоичных знаков, которые потребуются для кодирования слова МАГИЯ равно

$$3 + 3 + 3 + 3 + 3 = 15.$$

Ответ: 15.

Пример 2.

Для кодирования букв Е, Д, Г, О, Л решили использовать двоичное представление чисел 0, 1, 2, 3 и 4 соответственно (с сохранением одного незначащего нуля в случае одноразрядного представления). Закодируйте последовательность букв ГОЛОЛЕД таким способом и результат запишите шестнадцатеричным кодом.

Решение.

Представим данные в условии числа в двоичном коде:

Е	Д	Г	О	Л
0	1	2	3	4
00	01	10	11	100

Затем закодируем последовательность букв: ГОЛОЛЕД – 1011100111000001. Теперь разобьём это представление на четвёрки справа налево и переведем полученный набор чисел в шестнадцатеричный код.

$$1011\ 1001\ 1100\ 0001_2 - B9C1_{16}$$

Ответ: B9C1₁₆.

Пример 3.

Для 6 букв латинского алфавита заданы их двоичные коды (для некоторых букв из двух бит, для некоторых – из трех). Эти коды представлены в таблице:

A	B	C	D	E	F
00	100	10	011	11	101

Какая последовательность из 6 букв закодирована двоичной строкой 011111000101100?

Примечание: При решении этой задачи необходимо учитывать условия Фано и обратное условие Фано. Условие Фано означает, что ни одно кодовое слово не является началом другого кодового слова. Обратное условие Фано означает, что ни одно кодовое слово не является окончанием другого кодового слова

Решение.

Мы видим, что условия Фано и обратное условие Фано не выполняются, следовательно, код можно декодировать неоднозначно.

Рассмотрим различные варианты:

1) 011 11 100 0101100

Первая буква определяется однозначно, её код 011: D. Вторая буква также определяется однозначно – E. Определим для третьей буквы код 001 – B. Тогда следующая буква должна начинаться с кода 010, но таких букв в таблице нет, следовательно, этот вариант не подходит.

2) 011 11 10 00 101 100

В данном разбиении все буквы из таблицы найдены. Первая буква – D, вторая буква – E, третья буква – C. Далее следуют A, F и B.

Окончательно получили ответ: **DECAFB.**

Пример 4. Для кодирования некоторой последовательности, состоящей из букв А, Б, В и Г, решили использовать неравномерный двоичный код, позволяющий однозначно декодировать двоичную последовательность, появляющуюся на приёмной стороне канала связи. Для букв А, Б, В используются такие кодовые слова: А – 010, Б – 1, В – 011.

Укажите кратчайшее кодовое слово для буквы Г, при котором код будет допускать однозначное декодирование. Если таких кодов несколько, укажите код с **наименьшим** числовым значением.

Решение.

Рассмотрим два варианта решения поставленной задачи.

1 вариант решения.

Код не может начинаться с 1, так как буква Б имеет кодовое слово, равное 1.

Кодовое слово 0 не подходит, так как кодовые слова букв А и В начинаются с 0.

Кодовое слово 00 не включает в себя никакой из кодов и также не является подстрокой какого-либо кода, поэтому может быть определено для буквы Г.

2 вариант решения.

Для нахождения кодового слова наименьшей длины построим дерево (рис. 4.1). Влево будем откладывать нули, а вправо — единицы:

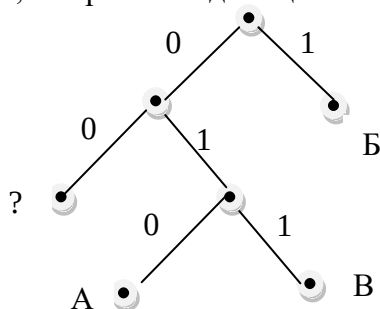


Рисунок 4.1 Дерево решений.

Поскольку у нас все ветви завершены листьями, т.е. буквами, кроме одной ветви, то остается единственный вариант, куда можно поставить букву Г: 00

ОТВЕТ: 00.

2) Вычисление количества различных сообщений в алфавите заданной мощности.

При решении задач этого типа необходимо знать понятия: алфавит, мощность алфавита, сообщение. Также необходимо знать правила вычисления количества сообщений при равновероятном и при неравновероятном вхождениях символов в сообщение.

Алфавит – это набор символов, используемый в том или ином языке для представления информации.

Мощность алфавита (N) – это полное количество символов в алфавите. Например, мощность алфавита, состоящего из букв А, П, С, Т, У равна 5.

Сообщение – это любая последовательность символов какого-либо алфавита, например, СТ, СТУ.

Количество различных сообщений при равновероятных вхождениях символов в сообщение вычисляется по формуле: $Q = N^k$.

где Q – количество различных сообщений;

N – мощность алфавита или количество используемых символов;

k – длина сообщения или количество символов в одном сообщении.

Задания, в которых требуется найти количество сообщений при неравновероятном (различном) вхождении (встречаемости) символов, можно решить способом перебора всевозможных вариантов либо с использованием формул комбинаторики.

Число сочетаний из n элементов по k элементов:

$$C_n^k = \frac{n!}{k!(n-k)!}$$

Факториал числа n : $n! = 1 \times 2 \times 3 \times \dots \times n$.

Следует отметить, что если слово состоит из k букв, причем есть m_1 вариантов выбора первой буквы, m_2 вариантов выбора второй буквы и т.д., то число возможных сообщений вычисляется как произведение: $M = m_1 \times m_2 \times \dots \times m_k$.

Рассмотрим примеры задач вычисления количества сообщений в алфавитах разной мощности.

Пример 1. Сколько существует всевозможных трехбуквенных слов в английском языке?

Решение:

В английском алфавите 26 букв, следовательно, мощность алфавита = 26. Длина сообщения $k = 3$. Найдем по формуле $Q = N^k$ количество трехбуквенных слов: $Q = 26^3$ или $\underline{26} \times \underline{26} \times \underline{26} = 17576$

Ответ: 17576 слов.

Пример 2. Сколько существует всевозможных *четырёхбуквенных* слов в алфавите из 4 букв: A, B, V, G , если известно, что буква A встречается *ровно два раза*?

Решение:

Длина сообщения = 4. Мощность алфавита = 4. Но мешает условие: буква A встречается *ровно два раза*. Подсчитаем, сколько слов можно составить, если буква A стоит на первом и втором местах. Тогда оставшиеся три буквы могут располагаться на третьем и четвертом месте. Такой вариант размещения букв дает $A A 3 3 = 3 \times 3 = 3^2 = 9$ четырехбуквенных слова. Аналогичный результат будет и для других размещений буквы A .

а) Найдем количество вариантов методом перебора:

$A A 3 3 = 9$ слов

$A 3 A 3 = 9$ слов

$A 3 3 A = 9$ слов

$3 A A 3 = 9$ слов

$3 A 3 A = 9$ слов

$3 3 A A = 9$ слов

Получили 6 вариантов. Каждый вариант содержит 9 четырехбуквенных слова.

Результат: Общее количество слов равно $6 \times 9 = 54$.

б) Найдем количество вариантов при помощи формулы числа сочетаний

$$C_N^k = \frac{N!}{k!(N-k)!}.$$

В задаче: $N = 4$, $k = 2$.

$$C_4^2 = \frac{4!}{2!(4-2)!} = \frac{1 \cdot 2 \cdot 3 \cdot 4}{1 \cdot 2 \cdot 1 \cdot 2} = 6$$

С учетом того, что каждый вариант содержит 9 четырехбуквенных слова, общее количество всех слов равно $6 \times 9 = 54$.

Оба подхода к решению дали одинаковый результат.

Ответ: 54 слова.

Пример 3. Шифр кодового замка представляет собой последовательность из пяти символов, каждый из которых является цифрой от 1 до 6.

Сколько различных шифров кодового замка можно задать, если известно, что цифра 1 должна встречаться в коде ровно 1 раз, а каждая из других допустимых цифр может встречаться в шифре любое количество раз или не встречаться совсем?

Решение:

Длина сообщения (k) = 5 символов. Мощность алфавита (N) = 6 (цифры от 1 до 6). Но так как цифра 1 встречается по условию ровно один раз, а остальные 5 цифр – любое количество раз, то будем считать, что $N = 5$ (цифры от 2 до 6, исключая 1). Рассмотрим вариант, когда цифра 1 стоит на первом месте, а остальные 5 цифр размещены на оставшихся 4 позициях: $1\ 5\ 5\ 5\ 5 = 1 \cdot 5^4 = 625$. Таким образом, один вариант размещения дает 625 шифров.

а) Найдем количество вариантов методом перебора:

$$\begin{aligned} 1\ 5\ 5\ 5\ 5 &= 625 \\ 5\ 1\ 5\ 5\ 5 &= 625 \\ 5\ 5\ 1\ 5\ 5 &= 625 \\ 5\ 5\ 5\ 1\ 5 &= 625 \\ 5\ 5\ 5\ 5\ 1 &= 625 \end{aligned}$$

получили 5 вариантов.

б) Найдем количество вариантов при помощи формулы числа сочетаний

$$C_N^k = \frac{N!}{k!(N-k)!}.$$

В задаче: $N = 5$, $k = 4$.

$$C_5^4 = \frac{5!}{4!(5-4)!} = \frac{1 \cdot 2 \cdot 3 \cdot 4 \cdot 5}{1 \cdot 2 \cdot 3 \cdot 4 \cdot 1} = 5$$

Вновь получили 5 вариантов.

С учетом того, что каждый вариант включает 625 шифров, вычислим общее количество шифров: $625 \times 5 = 3125$

Ответ: 3125.

Пример 4. Олег составляет таблицу кодовых слов для передачи сообщений, каждому сообщению соответствует своё кодовое слово. В качестве кодовых слов Олег

использует 4-буквенные слова, в которых есть только буквы А, Б, В, Г, Д и Е, причём буква Г появляется ровно 1 раз и только на первом или последнем месте. Каждая из других допустимых букв может встречаться в кодовом слове любое количество раз или не встречаться совсем.

Сколько различных кодовых слов может использовать Олег?

Решение:

Для получения количества возможных вариантов слов при одном из размещений буквы Г воспользуемся формулой $M = m_1 \times m_2 \times \dots \times m_k$,

где m_1 – количество вариантов выбора первой буквы, m_2 – количество вариантов выбора второй буквы и т.п.

Рассмотрим варианты, когда буква Г встречается на первом или последнем месте:

$$Г ? ? ? = 1 \times 5 \times 5 \times 5 = 5^3 = 125$$

$$? ? ? Г = 5 \times 5 \times 5 \times 1 = 5^3 = 125$$

Вместо знаков ? может стоять одна из пяти букв (А, Б, В, Д, Е), т.к. буква Г там стоять не может

Теперь суммируем количество найденных вариантов:

$$125 + 125 = 250$$

Ответ: 250.

4.2.2. Задания для самостоятельной подготовки

1. Для кодирования некоторой последовательности, состоящей из букв А, Б, В, Г, Д, Е, решили использовать неравномерный двоичный код, удовлетворяющий условию Фано. Для буквы А использовали кодовое слово 0, для буквы Б – кодовое слово 10.

Какова **наименьшая** возможная сумма длин кодовых слов для букв В, Г, Д, Е?

Примечание. Условие Фано означает, что никакое кодовое слово не является началом другого кодового слова. Это обеспечивает возможность однозначной расшифровки закодированных сообщений.

2. Для кодирования некоторой последовательности, состоящей из букв А, Б, В, Г, Д решили использовать неравномерный двоичный код, позволяющий однозначно декодировать полученную двоичную последовательность. Для буквы А использовали кодовое слово 01, для буквы Б – кодовое слово 101, для буквы В – 11, для буквы Г – 00.

Каким кодовым словом должна кодироваться буква Д? Если таких вариантов несколько, укажите самый короткий.

3. По каналу связи передаются сообщения, содержащие только семь букв: А, Б, И, К, Л, О, С. Для передачи используется двоичный код, удовлетворяющий условию Фано. Кодовые слова для некоторых букв известны: А – 001, И – 01 С – 10.

Какое **наименьшее** количество двоичных знаков потребуется для кодирования слова КОЛОБОК?

Примечание. Условие Фано означает, что ни одно кодовое слово не является началом другого кодового слова.

4. По каналу связи передаются сообщения, содержащие только семь букв: А, Б, Г, И, Н, Р, Т. Для передачи используется двоичный код, удовлетворяющий условию Фано. Кодовые слова для некоторых букв известны: Г – 110, И – 01, Т – 10.

Какое **наименьшее** количество двоичных знаков потребуется для кодирования слова БАРАБАН?

Примечание. Условие Фано означает, что ни одно кодовое слово не является началом другого кодового слова.

5. Для кодирования некоторой последовательности, состоящей из букв А, Б, В, Г, Д, Е, К решили использовать неравномерный двоичный код, удовлетворяющий условию Фано. Для буквы А использовали кодовое слово 1; для буквы Б – кодовое слово 01.

Какова наименьшая возможная сумма длин кодовых слов для букв В, Г, Д, Е, К?

Примечание. Условие Фано означает, что ни одно кодовое слово не является началом другого кодового слова.

6. По каналу связи передаются сообщения, содержащие только восемь букв: А, Е, И, О, П, Р, С, Т. Для передачи используется двоичный код, удовлетворяющий условию Фано. Кодовые слова для некоторых букв известны: А – 00, И – 1100, Р – 1110.

Какое наименьшее количество двоичных знаков потребуется для кодирования слова РЕПЕТИТОР?

Примечание. Условие Фано означает, что ни одно кодовое слово не является началом другого кодового слова.

7. Вася составляет 5-буквенные слова, в которых есть только буквы З, И, М, А, причём в каждом слове есть ровно одна гласная буква и она встречается ровно раз. Каждая из допустимых согласных букв может встречаться в слове любое количество раз или не встречаться совсем. Словом считается любая допустимая последовательность букв, не обязательно осмысленная.

Сколько существует таких слов, которые может написать Вася?

8. Для передачи закодированных сообщений используется таблица кодовых слов, в которой каждому сообщению соответствует кодовое слово из четырех букв. Причем используются только буквы Б, У и М.

Сколько различных кодовых слов может быть в такой таблице, если ни в одном слове нет трех одинаковых букв идущих подряд?

9. Для передачи закодированных сообщений используется таблица кодовых слов, в которой каждому сообщению соответствует кодовое слово из четырех букв. Причем используются только буквы А, Н, Б и Я.

Сколько различных кодовых слов может быть в такой таблице, если в каждом слове ровно один раз используется буква А?

10. Друзья решили составить таблицу кодовых слов для передачи секретных сообщений. Каждому сообщению соответствует определённое кодовое слово из шести букв.

Сколько различных кодовых слов друзья могут использовать, если в словах есть только буквы А, Б и В, причем буква А используется в каждом слове, но не больше одного раза?

11. Юля составляют таблицу кодовых слов для передачи сообщений, каждому сообщению соответствует своё кодовое слово. В качестве кодовых слов Юля решила использовать 6-буквенные слова, в которых есть только буквы Я, Н, Д, Е, К, С причём буква Я появляется ровно 2 раза. Каждая из других допустимых букв может встречаться в кодовом слове любое количество раз или не встречаться совсем.

Сколько различных кодовых слов может использовать Юля?

12. Оксана составляет таблицу кодовых слов для передачи сообщений, каждому сообщению соответствует своё кодовое слово. В качестве кодовых слов Оксана решила использовать 4-буквенные слова, в которых есть только буквы К, И, Н, О, причём буквы К и Н появляются ровно 1 раз. Каждая из других допустимых букв может встречаться в кодовом слове любое количество раз или не встречаться совсем.

Сколько различных кодовых слов может использовать Оксана?

13. Все пятибуквенные слова, составленные из букв Ф, А, Й, Л, записаны в алфавитном порядке и пронумерованы, начиная с 1. (Словом считается последовательность букв, необязательно осмысленная.) Начало списка выглядит так:

1. ААААА
2. ААААЙ
3. ААААЛ
4. ААААФ
5. АААЙА
6. АААЙЙ

...

Под каким номером в списке идёт первое слово, которое начинается с букв ЛФ?

14. Для передачи сообщений используются 5-буквенные коды. В кодах разрешается использовать только буквы Я, Н, Д, Е, К, С, при этом код не может начинаться с согласной и содержит ровно две гласные буквы. Каждая из допустимых букв может повторяться в коде несколько раз или не встречаться вовсе.

Сколько различных кодов удовлетворяют указанным условиям?

15. Все 4-буквенные слова, в составе которых могут быть буквы Н, О, Т, К, И, записаны в алфавитном порядке и пронумерованы, начиная с 1. Ниже приведено начало списка.

1. ИИИИ
2. ИИИК
3. ИИИН
4. ИИИО
5. ИИИТ
6. ИИКИ

Под каким номером в списке идёт первое слово, которое начинается с буквы О?

16. Для передачи сообщений используются 8-буквенные кодовые слова, содержащие 3 буквы А, 3 буквы Б и 2 буквы В.

Определите общее количество различных кодовых слов, в которых буквы А не стоят рядом.

4.3 Основы логики

4.3.1. Общие понятия и примеры решения задач

В заданиях работы по информатике используются следующие соглашения:

- символ 1 используется для обозначения значения «истина» (истинного высказывания)
- -символ 0 для обозначения значения «ложь» (ложного высказывания)

Для выполнения заданий необходимо знать логические операции, тождества и законы алгебра логики.

1. Логические операции и их обозначения:

а) **отрицание** (инверсия) обозначается $F = \bar{A}$ либо $F = \neg A$;

A	$F = \neg A$
0	1
1	0

б) **умножение** (конъюнкция) $F = A \wedge B$;

A	B	$F = A \wedge B$
0	0	0
0	1	0
1	0	0
1	1	1

в) **сложение** (дизъюнкция) обозначается $F = A \vee B$;

A	B	$F = A \vee B$
0	0	0
0	1	1
1	0	1
1	1	1

г) **следование** (импликация) обозначается $F = A \rightarrow B$;

A	B	$F = A \rightarrow B$
0	0	1
0	1	1
1	0	0
1	1	1

д) **равнозначность** (эквивалентность) обозначается $F = A \sim B$.

A	B	$F = A \sim B$
-----	-----	----------------

0	0	1
0	1	0
1	0	0
1	1	1

2. Приоритеты логических операций.

1. Операции в скобках.
2. Инверсия (отрицание).
3. Конъюнкция (логическое умножение).
4. Дизъюнкция (логическое сложение).
5. Импликация (следование).
6. Эквиваленция (равнозначность).

3. Тождества:

а) двойного отрицания $\neg(\neg A) = A$;

б) логического сложения;

$$\begin{aligned} A \vee 0 &= A & A \vee A &= A \\ A \vee 1 &= 1 & A \vee \neg A &= 1 \end{aligned}$$

в) логического умножения.

$$\begin{aligned} A \wedge 0 &= 0 & A \wedge A &= A \\ A \wedge 1 &= A & A \wedge \neg A &= 0 \end{aligned}$$

4. Законы алгебры логики:

а) коммутативный (переместительный) закон;

1. $A \vee B = B \vee A$
2. $A \wedge B = B \wedge A$

б) ассоциативный (сочетательный) закон;

1. $(A \vee B) \vee C = A \vee (B \vee C)$
2. $(A \wedge B) \wedge C = A \wedge (B \wedge C)$

в) дистрибутивный (распределительный) закон;

1. $(A \vee B) \wedge C = (A \wedge C) \vee (B \wedge C)$
2. $(A \wedge B) \vee C = (A \vee C) \wedge (B \vee C)$

с) законы де Моргана.

1. $\neg(A \vee B) = \neg A \wedge \neg B$
2. $\neg(A \wedge B) = \neg A \vee \neg B$
3. $A \rightarrow B = \neg A \vee B$
4. $A \sim B = (A \wedge B) \vee (\neg A \wedge \neg B)$

Рассмотрим в качестве примеры применения законов логики преобразование логического выражения.

Пример 1.

Пусть нам необходимо упростить логическое выражение:

$$(A \wedge B) \vee (A \wedge \neg B).$$

Решение.

Воспользуемся законом дистрибутивности и вынесем за скобки A :

$$(A \wedge B) \vee (A \wedge \neg B) = A \wedge (B \vee \neg B).$$

По закону логического сложения $B \vee \neg B = 1$, следовательно:

$$(A \wedge B) \vee (A \wedge \neg B) = A \wedge 1 = A.$$

Пример 2.

Для какого из указанных значений X истинно высказывание $\neg((X > 3) \rightarrow (X > 4))$?

1) 1 2) 2 3) 3 4) 4

Решение.

Определим $A = (X > 3)$, $B = (X > 4)$.

Тогда исходное выражение запишется как $\neg(A \rightarrow B)$. Упростим выражение, применив законы де Моргана.

$$1. \quad A \rightarrow B = \neg A \vee B$$

$$2. \quad \neg(\neg A \vee B) = A \wedge \neg B$$

В наших обозначениях $B = (X > 4)$, следовательно $\neg B = (X \leq 4)$. Ответом будет число, удовлетворяющее условию $(3 < X \leq 4)$. Из предложенных вариантов выбираем число 4.

Ответ: 4.

Пример 3.

В школе-новостройке в каждой из двух аудиторий может находиться либо кабинет информатики, либо кабинет физики. На дверях аудиторий повесили шуточные таблички. На первой повесили табличку «По крайней мере, в одной из этих аудиторий размещается кабинет информатики», а на второй аудитории — табличку с надписью «Кабинет физики находится в другой аудитории». Проверяющему, который пришел в школу, известно только, что надписи на табличках либо обе истинны, либо обе ложны. Помогите проверяющему найти кабинет информатики.

Решение.

Переведем условие задачи на язык логики высказываний. Так как в каждой из аудиторий может находиться кабинет информатики, то пусть:

A = «В первой аудитории находится кабинет информатики»;

B = «Во второй аудитории находится кабинет информатики».

Отрицания этих высказываний:

$\bar{A}$ = «В первой аудитории находится кабинет физики»;

$\bar{B}$ = «Во второй аудитории находится кабинет физики».

Высказывание, содержащееся на табличке на двери первой аудитории, соответствует логическому выражению:

$$X = A \vee B.$$

Высказывание, содержащееся на табличке на двери второй аудитории, соответствует логическому выражению:

$$Y = \bar{A}.$$

Содержащееся в условии задачи утверждение о том, что надписи на табличках либо одновременно истинны, либо одновременно ложны в соответствии с законом исключенного третьего записывается следующим образом:

$$(X \& Y) \vee (\bar{X} \& \bar{Y}) = 1$$

Подставим вместо X и Y соответствующие формулы:

$$(X \& Y) \vee (\bar{X} \& \bar{Y}) = ((A \vee B) \& \bar{A}) \vee ((A \vee B) \& \bar{\bar{A}}).$$

Упростим сначала первое слагаемое. В соответствии с законом дистрибутивности умножения относительно сложения:

$$(A \vee B) \& \bar{A} = A \& \bar{A} \vee B \& \bar{A}$$

В соответствии с законом непротиворечия:

$$A \& \bar{A} \vee B \& \bar{A} = 0 \vee B \& \bar{A}$$

Упростим теперь второе слагаемое. В соответствии с первым законом де Моргана и законом двойного отрицания:

$$\overline{(A \vee B) \& \bar{A}} = \bar{A} \& \bar{B} \& A = \bar{A} \& A \& \bar{B}.$$

В соответствии с законом непротиворечия:

$$\bar{A} \& A \& \bar{B} = 0 \& \bar{B} = 0.$$

В результате получаем:

$$(0 \vee B \& \bar{A}) \vee 0 = B \& \bar{A}$$

Полученное логическое выражение оказалось простым и поэтому его можно проанализировать без построения таблицы истинности. Для того чтобы выполнялось равенство $B \& \bar{A} = 1$, B и $\bar{A}$ должны быть равны 1, то есть соответствующие им высказывания истинны.

Ответ: В первой аудитории находится кабинет физики, а во второй — кабинет информатики.

Пример 4.

На числовой прямой даны два отрезка: $P = [2, 10]$ и $Q = [6, 14]$. Выберите такой отрезок A , что формула

$$((x \in A) \rightarrow (x \in P)) \vee (x \in Q)$$

тождественно истинна, то есть принимает значение 1 при любом значении переменной x .

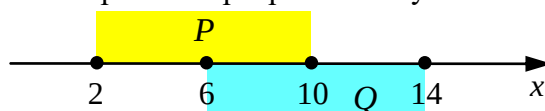
- 1) $[0, 3]$ 2) $[3, 11]$ 3) $[11, 15]$ 4) $[15, 17]$

Решение (вариант 1):

1. Два условия связаны с помощью операции $\vee$ («ИЛИ»), поэтому должно выполняться хотя бы одно из них. Для того, чтобы упростить понимание выражения, обозначим отдельные высказывания буквами: **A**: $x \in A$, **P**: $x \in P$, **Q**: $x \in Q$. Тогда получаем, переходя к более простым обозначениям: $Z = (A \rightarrow P) + Q$

2. Представим импликацию $A \rightarrow P$ через операции «ИЛИ» и «НЕ»: $A \rightarrow P = \bar{A} + P$, так что получаем $Z = \bar{A} + P + Q$. Это значит, что для тождественной истинности выражения Z нужно, чтобы для любого x было выполнено одно из условий: $\bar{A}$, **P**, **Q**; из всех этих выражений нам **неизвестно только $\bar{A}$** .

3. Посмотрим, какие интервалы перекрываются условиями **P** и **Q**:



На рисунке видим, что отрезок $[2, 14]$ перекрыт, поэтому выражение $\bar{A}$ должно перекрывать оставшуюся часть; таким образом, $\bar{A}$ должно быть истинно на интервалах $(-\infty, 2)$ и $(14, \infty)$ и, соответственно, выражение **A** (без инверсии) может быть истинно только внутри отрезка $[2, 14]$.

Из всех отрезков, приведенных в условии, только отрезок $[3, 11]$ (вариант 2) находится целиком внутри отрезка $[2, 14]$, это и есть правильный ответ

Ответ: 2.

Решение (вариант 2).

1. Пп 1-2 такие же, как и в предыдущем способе решения.
2. Если рассматривать все значения x на числовой прямой, то логические значения формул могут измениться только при переходе через граничные точки заданных промежутков. Эти точки (2, 6, 10 и 14) разбивают числовую прямую на несколько интервалов, для каждого из которых можно определить логическое значение выражения $Y = P + Q$

x	P	Q	$Y = P + Q$
$x < 2$	0	0	0
$2 < x < 6$	1	0	1
$6 < x < 10$	1	1	1
$10 < x < 14$	0	1	1
$x > 14$	0	0	0

Для упрощения записи не будем рассматривать значения формул на концах отрезков, так как это не влияет на решение.

3. По условию выражение $Z = \bar{A} + P + Q$ должно быть равно 1 при любых значениях x , то есть, в соответствующем столбце таблицы должны быть все единицы; отсюда можно найти, каким должно быть значение $\bar{A}$ (и соответствующее значение A) для каждого интервала:

x	P	Q	$Y = P + Q$	$\bar{A}$	A	$Z = \bar{A} + P + Q$
$x < 2$	0	0	0	1	0	1
$2 < x < 6$	1	0	1	любое	любое	1
$6 < x < 10$	1	1	1	любое	любое	1
$10 < x < 14$	0	1	1	любое	любое	1
$x > 14$	0	0	0	1	0	1

4. Таким образом, значение A должно быть равно 0 вне отрезка $[2, 14]$; из всех отрезков, приведенных в условии, только отрезок $[3, 11]$ (вариант 2)

Ответ: 2.

4.3.2. Задания для самостоятельной подготовки

1. На числовой прямой даны два отрезка: $P = [5, 15]$ и $Q = [12, 18]$. Выберите такой отрезок A , что формула

$$((x \in A) \rightarrow (x \in P)) \vee (x \in Q)$$

тождественно истинна, то есть принимает значение 1 при любом значении переменной x .

- 1) $[3, 11]$ 2) $[2, 21]$ 3) $[10, 17]$ 4) $[15, 20]$

Примечание: $\rightarrow$ — операция следования, $\in$ — принадлежит, $\vee$ — логическое ИЛИ.

2. На числовой прямой даны два отрезка: $P = [5, 10]$ и $Q = [15, 18]$. Выберите такой отрезок A , что формула

$$((x \in A) \rightarrow (x \in P)) \vee (x \in Q)$$

тождественно истинна, то есть принимает значение 1 при любом значении переменной x .

- 1) $[3, 11]$ 2) $[6, 10]$ 3) $[8, 16]$ 4) $[17, 23]$

Примечание: $\rightarrow$ — операция следования, $\in$ — принадлежит, $\vee$ — логическое ИЛИ.

3. На числовой прямой даны два отрезка: $P = [10, 20]$ и $Q = [12, 15]$. Выберите такой отрезок A , что формула

$$((x \notin A) \rightarrow (x \notin P)) \vee (x \in Q)$$

тождественно истинна, то есть принимает значение 1 при любом значении переменной x .

- 1) $[10, 15]$ 2) $[20, 35]$ 3) $[5, 20]$ 4) $[12, 40]$

Примечание: $\rightarrow$ — операция следования, $\in$ — принадлежит, $\vee$ — логическое ИЛИ.

4. На числовой прямой даны два отрезка: $P = [10, 20]$ и $Q = [15, 25]$. Выберите такой отрезок A , что формула

$$((x \in P) \rightarrow (x \in Q)) \vee (x \in A)$$

тождественно истинна, то есть принимает значение 1 при любом значении переменной x .

- 1) $[8, 17]$ 2) $[10, 12]$ 3) $[15, 22]$ 4) $[12, 18]$

Примечание: $\rightarrow$ — операция следования, $\in$ — принадлежит, $\vee$ — логическое ИЛИ.

5. Для какого наибольшего целого числа A формула

$$((x \leq 9) \rightarrow (x \times x \leq A)) \wedge ((y \times y \leq A) \rightarrow (y \leq 9))$$

тождественно истинна, то есть принимает значение 1 при любых целых неотрицательных x и y ?

6. Сколько существует целых значений числа A , при которых формула

$$((x < 6) \rightarrow (x^2 < A)) \wedge ((y^2 \leq A) \rightarrow (y \leq 6))$$

тождественно истинна при любых целых неотрицательных x и y ?

7. Сколько существует целых значений числа A , при которых формула

$$((x < 5) \rightarrow (x^2 < A)) \wedge ((y^2 \leq A) \rightarrow (y \leq 5))$$

тождественно истинна при любых целых неотрицательных x и y ?

8. Сколько существует целых значений числа A , при которых формула

$$((x < A) \rightarrow (x^2 < 100)) \wedge ((y^2 \leq 64) \rightarrow (y \leq A))$$

тождественно истинна при любых целых неотрицательных x и y ?

9. Сколько существует целых значений числа A , при которых формула

$$((x < A) \rightarrow (x^2 < 81)) \wedge ((y^2 \leq 36) \rightarrow (y \leq A))$$

тождественно истинна при любых целых неотрицательных x и y ?

10. Сколько различных решений имеет уравнение $J \wedge \neg K \wedge L \wedge \neg M \wedge (N \vee \neg N) = 0$, где J, K, L, M, N — логические переменные?

В ответе не нужно перечислять все различные наборы значений J, K, L, M и N , при которых выполнено данное равенство. В качестве ответа нужно указать количество таких наборов.

11. Сколько различных решений имеет уравнение

$$((J \rightarrow K) \rightarrow (M \wedge N \wedge L)) \wedge ((J \wedge \neg K) \rightarrow \neg (M \wedge N \wedge L)) \wedge (M \rightarrow J) = 1$$

где J, K, L, M, N — логические переменные?

В ответе не нужно перечислять все различные наборы значений J, K, L, M и N , при которых выполнено данное равенство. В качестве ответа нужно указать количество таких наборов.

12. Сколько различных решений имеет уравнение

$$((K \vee L) \rightarrow (L \wedge M \wedge N)) = 0$$

где K, L, M, N — логические переменные? В Ответе не нужно перечислять все различные наборы значений K, L, M и N , при которых выполнено данное равенство. В качестве Ответа Вам нужно указать количество таких наборов.

13. Сколько различных решений имеет уравнение

$$(X \wedge Y \vee Z) \rightarrow (Z \vee P) = 0$$

где X, Y, Z, P — логические переменные? В ответе не нужно перечислять все различные наборы значений, при которых выполнено данное равенство. В качестве ответа вам нужно указать только количество таких наборов.

14. Сколько различных решений имеет уравнение

$$((A \rightarrow B) \wedge C) \vee (D \wedge \neg D) = 1,$$

где A, B, C, D — логические переменные?

В ответе не нужно перечислять все различные наборы значений A, B, C, D , при которых выполнено данное равенство. В качестве ответа вам нужно указать количество таких наборов.

4.4 Элементы теории алгоритмов. Задания для самостоятельной подготовки

Для успешного выполнения заданий этой темы необходимо иметь навыки формально исполнения алгоритма, умения читать и понимать программы, написанные на алгоритмических языках (C, Pascal, Python, Basic).

1. 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.

2. Автомат получает на вход четырёхзначное десятичное число, в котором все цифры разные. По этому числу строится новое число по следующим правилам.

1. Вычисляются сумма наибольшей и наименьшей цифр, а также сумма оставшихся цифр.
2. Полученные суммы записываются в порядке невозрастания (без разделителей).

Пример. Исходное число: 1263. $1 + 6 = 7$; $2 + 3 = 5$. Результат: 75.

Укажите **наименьшее** число, при обработке которого автомат выдаёт результат 117.

3. Автомат обрабатывает натуральное число $N > 1$ по следующему алгоритму.

1. Строится двоичная запись числа N .
2. Последняя цифра двоичной записи удаляется.
3. Если исходное число N было нечётным, в конец записи (справа) дописываются цифры 10, если чётным – 01.
4. Результат переводится в десятичную систему и выводится на экран.

Пример. Дано число 13. Алгоритм работает следующим образом.

1. Двоичная запись числа N : 1101.
2. Удаляется последняя цифра, новая запись: 110.
3. Исходное число нечётно, дописываются цифры 10, новая запись: 11010.
4. На экран выводится число 26.

Какое число нужно ввести в автомат, чтобы в результате получилось 2018?

4. Автомат обрабатывает натуральное число N ($0 \leq N \leq 255$) по следующему алгоритму:

1. Строится восьмибитная двоичная запись числа N .
2. Все цифры двоичной записи заменяются на противоположные (0 на 1, 1 на 0).
3. Полученное число переводится в десятичную запись.

4. Из нового числа вычитается исходное, полученная разность выводится на экран.

Пример. Дано число $N=13$. Алгоритм работает следующим образом:

1. Восьмибитная двоичная запись числа N : 00001101.
2. Все цифры заменяются на противоположные, новая запись 11110010.
3. Десятичное значение полученного числа 242.
4. На экран выводится число $242 - 13 = 229$.

Какое число нужно ввести в автомат, чтобы в результате получилось 111?

5. Автомат обрабатывает натуральное число N ($0 \leq N \leq 255$) по следующему алгоритму:

1. Строится восьмибитная двоичная запись числа N .
2. Все цифры двоичной записи заменяются на противоположные (0 на 1, 1 на 0).
3. Полученное число переводится в десятичную запись.
4. Из нового числа вычитается исходное, полученная разность выводится на экран.

ран.

Какое число нужно ввести в автомат, чтобы в результате получилось 133?

6. Запишите число, которое будет напечатано в результате выполнения следующей программы.

Бейсик	Python
<pre>DIM S, N AS INTEGER S = 0 N = 75 WHILE (S + N) < 150 S = S + 15 N = N - 5 WEND PRINT N</pre>	<pre>s = 0 n = 75 while (s + n) < 150: s = s + 15 n = n - 5 print(n)</pre>
Алгоритмический язык	Паскаль
<pre>алг нач цел n, s s := 0 n := 75 нц пока s + n < 150 s := s + 15 n := n - 5 кц вывод n кон</pre>	<pre>var s, n: integer; begin s := 0; n := 75; while (s + n) < 150 do begin s := s + 15; n := n - 5; end; writeln(n) end.</pre>
C++	
<pre>#include <iostream> using namespace std;</pre>	

```
int main() {
    int s = 0, n = 75;
    while (s + n < 150) {
        s = s + 15;
        n = n - 5;
    }
    cout << n << endl;
    return 0;
}
```

7. Запишите число, которое будет напечатано в результате выполнения следующей программы.

Pascal	Basic
<pre>var s, n : longint; begin s := -1; n := 3; while 6*s*s*s*s*s > 800*s*s*s do begin s := s - 2; n := n * 2 end; write(n - s) end.</pre>	<pre>Dim S, N As Long S = -1 N = 3 While 6*S*S*S*S*S > 800*S*S*S S = S - 2 N = N * 2 Wend Print N - S</pre>
C++	Python
<pre>#include <iostream> void main() { int s = -1, n = 3; while (6*s*s*s*s*s > 800*s*s*s) { s -= 2; n *= 2; } std::cout << n - s; return 0; }</pre>	<pre>s, n = -1, 3 while 6*s*s*s*s*s > 800*s*s*s : s -= 2 n *= 2 print(n - s)</pre>

8. Запишите число, которое будет напечатано в результате выполнения следующей программы. Для Вашего удобства программа представлена на пяти языках программирования.

Бейсик	Python
<pre>DIM S, N AS INTEGER S = 0</pre>	<pre>s = 0 n = 40 while s + n < 100:</pre>

<pre> N = 40 WHILE S + N < 100 S = S + 20 N = N - 5 WEND PRINT S </pre>	<pre> s = s + 25 n = n - 5 print(s) </pre>
Алгоритмический язык	Паскаль
<pre> нач цел s, n s := 0 n := 40 нц пока s + n < 100 s := s + 25 n := n - 5 кц вывод s кон </pre>	<pre> var s, n: integer; begin s := 0; n := 40; while s + n < 100 do begin s := s + 25; n := n - 5 end; writeln(s) end. </pre>
C++	
<pre> #include <iostream> using namespace std; int main() { int s = 0, n = 40; while (s + n < 100) { s = s + 25; n = n - 5; } cout << s; return 0; } </pre>	

9. Укажите наибольшее число x , при вводе которого алгоритм печатает сначала 21, а потом 3.

Бейсик	Python
<pre> DIM X, L, M AS INTEGER INPUT X L = 1 M = 0 WHILE X > 0 M = M + 1 IF X MOD 2 <> 0 THEN L = L * (X MOD 8) END IF X = X \ 8 WEND PRINT L </pre>	<pre> x = int(input()) L = 1 M = 0 while x > 0: M = M + 1 if x % 2 != 0: L = L * (x % 8) x = x // 8 print(L) print(M) </pre>

PRINT M	
Алгоритмический язык	Паскаль
алг нач цел x, L, M ввод x L := 1 M := 0 нц пока x > 0 M := M + 1 если mod(x,2) <> 0 то L := L * mod(x,8) все x := div(x,8) кц вывод L, M кон	var x, L, M: integer; begin readln(x); L := 1; M := 0; while x > 0 do begin M := M + 1; if x mod 2 <> 0 then L := L * (x mod 8); x := x div 8 end; writeln(L); writeln(M) end.
C++	
<pre>#include <iostream> using namespace std; int main(){ int x, L, M; cin >> x; L = 1; M = 0; while (x > 0) { M = M + 1; if(x % 2 != 0) { L = L * (x % 8); } x = x / 8; } cout << L << endl << M << endl; return 0; }</pre>	

10. Ниже на пяти языках записан алгоритм, который получает на вход число x и печатает два числа a и b .

Бейсик	Python
DIM X, A, B AS INTEGER INPUT X A=0: B=1 WHILE (X MOD 10) > 1 A = A+(X MOD 10) B = B*2 X = X\10 WEND PRINT A PRINT B	<pre>x=int(input()) a=0 b=1 while (x%10)>1: a=a+(x%10) b=b*2 x=x//10 print(a) print(b)</pre>

Алгоритмический язык	Паскаль
алг нач цел x, a, b ввод x a:=0; b:=1 нц пока mod(x,10)>1 a:=a+mod(x,10) b:=b*2 x:=div(x,10) кц вывод a, nc, b кон	var x, a, b: integer; begin readln(x); a:=0; b:=1; while (x mod 10) > 1 do begin a:=a+(x mod 10); b:=b*2; x:=x div 10; end; writeln(a); write(b); end.
C++	
<pre>#include <stdio.h> void main() { int x, a, b; scanf("%d", &x); a=0; b=1; while ((x%10)>1){ a=a+(x%10); b=b*2; x=x/10; } printf("%d\n%d", a, b); return 0; }</pre>	

Укажите наименьшее из таких чисел x , при вводе которых алгоритм сначала печатает 12, а потом 16.

11. Ниже на четырех языках записан алгоритм, который получает на вход число x и печатает два числа L и M .

Бейсик	Python
DIM X, L, M AS INTEGER INPUT X L = 1 M = 0 WHILE X > 0 M = M + 1 IF X MOD 2 <> 0 THEN L = L * (X MOD 8) END IF X = X \ 8 WEND PRINT L PRINT M	<pre>x = int(input()) L = 1 M = 0 while x > 0: M = M + 1 if x % 2 != 0: L = L * (x % 8) x = x // 8 print(L) print(M)</pre>
C++	Паскаль
#include <iostream>	var x, L, M: integer;

<pre>using namespace std; int main() {int x, L, M; cin >> x; L = 1; M = 0; while (x > 0) { M = M + 1; if(x % 2 != 0) { L = L * (x % 8); } x = x / 8; } cout << L << endl << M << endl; return 0; }</pre>	<pre>begin readln(x); L := 1; M := 0; while x > 0 do begin M := M + 1; if x mod 2 <> 0 then L := L * (x mod 8); x := x div 8 end; writeln(L); writeln(M) end.</pre>
---	--

Укажите наибольшее из таких чисел x , при вводе которых алгоритм сначала печатает 35, а потом 3.

12. В программе используется одномерный целочисленный массив A с индексами от 0 до 10. Ниже представлен записанный на разных языках программирования фрагмент этой программы.

Бейсик	Python
<pre>s = 0 n = 10 FOR i = 0 TO n-1 s = s+A(i)-A(i+1) NEXT i</pre>	<pre>s = 0 n = 10 for i in range(n): s = s + A[i] - A[i+1]</pre>
Паскаль	Си++
<pre>s := 0; n := 10; for i:=0 to n-1 do begin s := s + A[i] - A[i+1]; end;</pre>	<pre>s = 0; n = 10; for (i = 0; i < n; i++) { s = s + A[i] - A[i+1]; }</pre>

В начале выполнения этого фрагмента в массиве находились числа 27, 17, 7, 0, 7, 17, 27, 17, 10, 7, 0, т.е. $A[0] = 27$, $A[1] = 17$ и т.д. Чему будет равно значение переменной s после выполнения данного фрагмента программы?

13. 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 различных языках программирования).

Бейсик	Python
<pre>j = 5 WHILE A(j) < A(j-1)</pre>	<pre>j = 5 while A[j] < A[j-1]:</pre>

<pre> t = A(j) A(j) = A(j-1) A(j-1) = t j = j - 1 WEND </pre>	<pre> A[j],A[j-1]=A[j-1],A[j] j -= 1 </pre>
Паскаль	Си++
<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>	<pre> j = 5; while (A[j] < A[j-1]) { t = A[j]; A[j] = A[j-1]; A[j-1] = t; j -= 1; } </pre>

14. В программе описан одномерный целочисленный массив A с индексами от 0 до 10 и целочисленные переменные i и t . Ниже представлен фрагмент этой программы, записанный на разных языках программирования.

Бейсик	Паскаль
<pre> FOR i = 0 TO 10 A(i) = i-1 NEXT i FOR i = 0 TO 9 A(i) = A(i+1) NEXT i A(10) = 10 </pre>	<pre> for i := 0 to 10 do A[i] := i-1; for i := 0 to 9 do A[i] := A[i+1]; A[10] := 10; </pre>
Си++	Python
<pre> for (i = 0; i <= 10; i++) { A[i] = i-1; } for (i = 0; i <= 9; i++) { A[i] = A[i+1]; } A[10] = 10; </pre>	<pre> for i in range(0, 11): A[i] = i-1 for i in range(0, 10): A[i] = A[i+1] A[10] = 10 </pre>

Чему окажутся равны элементы этого массива после выполнения фрагмента программы?

- 1) 9 9 9 9 9 9 9 9 9 10
- 2) 1 2 3 4 5 6 7 8 9 10 10
- 3) -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 10
- 4) 0 1 2 3 4 5 6 7 8 9 10

15. В программе описан одномерный целочисленный массив A с индексами от 0 до 10 и целочисленные переменные i и t . Ниже представлен фрагмент этой программы, записанный на разных языках программирования.

Бейсик	Паскаль
<pre> FOR i = 0 TO 10 A(i) = i NEXT i t = A(10) FOR i = 10 TO 1 STEP -1 </pre>	<pre> for i := 0 to 10 do A[i] := i; t := A[10]; for i := 10 downto 1 do A[i] := A[i-1]; </pre>

<pre> A(i) = A(i-1) NEXT i A(0) = t </pre>	<pre> A[0] := t; </pre>
Си++	Python
<pre> for (i = 0; i <= 10; i++) { A[i] = i; } t = A[10]; for (i = 10; i >= 1; i--) { A[i] = A[i-1]; } A[0] = t; </pre>	<pre> for i in range(0, 11): A[i] = i t = A[10] for i in range(10, 0, -1): A[i] = A[i-1] A[0] = t </pre>

Чему окажутся равны элементы этого массива после выполнения фрагмента программы?

- 1) 10 10 10 10 10 10 10 10 10 10 10
- 2) 10 0 1 2 3 4 5 6 7 8 9
- 3) 10 0 0 0 0 0 0 0 0 0 0
- 4) 10 2 3 4 5 6 7 8 9 10 10

16. В программе описан одномерный целочисленный массив A с индексами от 0 до 9 и целочисленные переменные i и t . Ниже представлен фрагмент этой программы, записанный на разных языках программирования.

Бейсик	Паскаль
<pre> FOR i = 0 TO 9 A(i) = i+1 NEXT i t = A(0) FOR i = 1 TO 9 A(i-1) = A(i) NEXT i A(9) = t </pre>	<pre> for i := 0 to 9 do A[i] := i+1; t := A[0]; for i := 1 to 9 do A[i-1] := A[i]; A[9] := t; </pre>
Си++	Python
<pre> for (i = 0; i <= 9; i++) { A[i] = i+1; } t = A[0]; for (i = 1; i <= 9; i++) { A[i-1] = A[i]; } A[9] = t; </pre>	<pre> for i in range(0, 10): A[i] = i+1 t = A[0] for i in range(1, 10): A[i-1] = A[i] A[9] = t </pre>

Чему окажутся равны элементы этого массива после выполнения фрагмента программы?

- 1) 1 2 3 4 5 6 7 8 9 1
- 2) 1 1 1 1 1 1 1 1 1 1
- 3) 10 10 10 10 10 10 10 10 10 1
- 4) 2 3 4 5 6 7 8 9 10 1

17. Определите число, которое будет напечатано в результате выполнения следующего алгоритма, написанного ниже на четырех языках программирования.

Примечание. Функция *abs* возвращает абсолютное значение своего входного параметра.

Бейсик	Паскаль
<pre> DIM A, B, T, M, R AS LONG A = -20: B = 20 M = A: R = F(A) FOR T = A TO B IF F(T) <= R THEN M = T R = F(T) END IF NEXT T PRINT M + R FUNCTION F(x) F = abs(abs(x - 6) + abs(x + 6) - 16) + 2 END FUNCTION </pre>	<pre> var a, b, t, M, R : longint; function F(x: longint) : longint; begin F := abs(abs(x - 6) + abs(x + 6) - 16) + 2; end; begin a := -20; b := 20; M := a; R := F(a); for t := a to b do begin if (F(t) <= R) then begin M := t; R := F(t) end end; write(M + R) end. </pre>
Си++	Python
<pre> #include <iostream> using namespace std; long F(long x) { return abs(abs(x - 6) + abs(x + 6) - 16) + 2; } int main() { long a = -20, b = 20, M = a, R = F(a); for (int t = a; t <= b; ++t) { if (F(t) <= R) { M = t; R = F(t); } } cout << M + R; return 0; } </pre>	<pre> def F(x): return abs(abs(x - 6) + abs(x + 6) - 16) + 2 a = -20 b = 20 M = a R = F(a) for t in range(a, b + 1): if (F(t) <= R): M = t R = F(t) print (M + R) </pre>

18. Определите число, которое будет напечатано в результате выполнения следующего алгоритма, написанного ниже на четырех языках программирования.

Бейсик	Паскаль
<pre> DIM T, M AS INTEGER M = -5 FOR T = -4 TO 5 IF F(T) > F(M) THEN M = T END IF </pre>	<pre> function f(x : integer) : integer; begin if x > 2 then f := x * x - 23 else f := x * x + 3 * x - 7 end end </pre>

<pre> NEXT T PRINT 5 - M FUNCTION F(X) IF X > 2 THEN F = X * X - 23 ELSE F = X * X + 3 * X - 7 END IF END FUNCTION </pre>	<pre> end; var t, M : integer; begin M := -5; for t := -4 to 5 do if f(t) > f(M) then M := t; writeln(5 - M) end. </pre>
Си++	Python
<pre> #include <iostream> int f(int x){ if (x > 2) return x * x - 23; return x * x + 3 * x - 7; } int main(){ int M = -5; for (int t = -4; t < 6; t++) if (f(t) > f(M)) M = t; std::cout << 5 - M; return 0; } </pre>	<pre> def f(x) : if x > 2 : return x * x - 23 return x * x + 3 * x - 7 M = -5 for t in range(-4, 6) : if f(t) > f(M) : M = t print(5 - M) </pre>

19. Какое число будет напечатано в результате работы следующей программы?
Для Вашего удобства программа приведена на четырех языках программирования.

Бейсик	Паскаль
<pre> DIM A, B, T, M, R AS INTEGER A = -11: B = 11 M = A: R = F(A) FOR T = A TO B IF F(T) <= R THEN M = T R = F(T) END IF NEXT T PRINT M+R FUNCTION F(x) F = 2*(x*x-50)*(x*x-50)+6 </pre>	<pre> var a, b, t, M, R :integer; function F(x:integer):integer; begin F := 2*(x*x-50)*(x*x-50)+6 end; begin a := -11; b := 11; M := a; R := F(a); for t := a to b do begin if F(t) <= R then begin M := t; R := F(t) end end </pre>

END FUNCTION	end; write(M+R) end.
Си++	Python
<pre>#include <iostream> using namespace std; int F(int x) { return 2*(x*x-50)*(x*x-50)+6; } int main() { int a, b, t, M, R; a = -11; b = 11; M = a; R = F(a); for (t=a; t<=b; ++t) { if (F(t) <= R) { M = t; R = F(t); } } cout << M+R; return 0; }</pre>	<pre>def F(x): return 2*(x*x-50)*(x*x-50)+6 a=-11; b=11 M=a; R=F(a) for t in range(a,b+1): if F(t) <= R: M=t; R=F(t) print(M+R)</pre>

20. Какое число будет напечатано в результате работы следующей программы?
Для Вашего удобства программа приведена на четырех языках программирования.

Бейсик	Паскаль
<pre>DIM A, B, T, M, R AS INTEGER A = -11: B = 11 M = A: R = F(A) FOR T = A TO B IF F(T) <= R THEN M = T R = F(T) END IF NEXT T PRINT M+R FUNCTION F(x) IF X>0 THEN F = x*x + 4 ELSE F = x*x + 5 END IF END FUNCTION</pre>	<pre>var a, b, t, M, R :integer; function F(x:integer):integer; begin if x>0 then F := x*x + 4 else F := x*x + 5 end; begin a := -11; b := 11; M := a; R := F(a); for t := a to b do begin if F(t) <= R then begin M := t; R := F(t) end end; write(M+R) end.</pre>
Си++	Python
<pre>#include <iostream> using namespace std;</pre>	<pre>def F(x): if x>0:</pre>

<pre> int F(int x) { if (x>0) return x*x + 4; else return x*x + 5; } int main() { int a, b, t, M, R; a = -11; b = 11; M = a; R = F(a); for (t=a; t<=b; ++t) { if (F(t) <= R) { M = t; R = F(t); } } cout << M+R; return 0; } </pre>	<pre> return x*x + 4 else: return x*x + 5 a=-11; b=11 M=a; R=F(a) for t in range(a,b+1): if F(t) <= R: M=t; R=F(t) print(M+R) </pre>
---	---

21. Какое число будет напечатано в результате работы следующей программы?
Для Вашего удобства программа приведена на четырех языках программирования.

Бейсик	Паскаль
<pre> DIM A AS LONG DIM T AS LONG DIM N AS LONG FUNCTION F (x AS INTEGER) AS LONG F = 2*(x-7)*(x-7)+7*x+7 END FUNCTION FUNCTION G (x AS INTEGER) AS LONG G = 3*x*x - 107 END FUNCTION A = 0: N = 20 FOR T = 1 TO 20 IF F(N) >= G(T) THEN A = A + 1 END IF NEXT T PRINT A </pre>	<pre> var n, a, t:longint; function F(x: longint): longint; begin F:= 2*(x-7)*(x-7)+7*x+7; end; function G(x: longint): longint; begin G:= 3*x*x - 107; end; begin n:=20; a:=0; for t:= 1 to 20 do begin if (F(n) >= G(t)) then a:=a+1; end; writeln(a); end. </pre>
Си++	Python
<pre> #include <iostream> using namespace std; long F(long x) {return 2*(x-7)*(x-7)+7*x+7; } </pre>	<pre> def F(x): return 2*(x-7)*(x-7)+7*x+7 def G(x): return 3*x*x - 107 </pre>

<pre> long G(long x) {return 3*x*x - 107; } int main() {long a = 0, n; n = 20; for (int t = 1; t <= 20; t++) {if (F(n) >= G(t)) a++; } cout << a << endl; return 0; } </pre>	<pre> n = 20 a = 0 for t in range(1,21): if (F(n) >= G(t)): a += 1 print (a) </pre>
---	--

22. Чему будет равно значение, вычисленное алгоритмом при выполнении вызова F(5)?

Бейсик	Python
<pre> SUB F(n) IF n > 2 THEN F = F(n - 1) + F(n-2) ELSE F = 1 END IF END SUB </pre>	<pre> def F(n): if n > 2: return F(n-1)+ F(n-2) else: return 1 </pre>
Паскаль	Си++
<pre> function F(n: integer): integer; begin if n > 2 then F := F(n - 1) + F(n - 2) else F := 1; end; end; </pre>	<pre> int F(int n) { if (n > 2) return F(n-1) + F(n-2); else return 1; } </pre>

23. Чему равна сумма всех чисел, напечатанных на экране при выполнении вызова F(3)?

Бейсик	Python
<pre> SUB F(n) PRINT n IF n < 5 THEN F(n + 1) F(n + 2) END IF END SUB </pre>	<pre> def F(n): print(n) if n < 5: F(n + 1) F(n + 2) </pre>
Си++	Паскаль

<pre>void F(int n) { cout << n; if (n < 5) { F(n + 1); F(n + 2); } }</pre>	<pre>procedure F(n: integer); begin writeln(n); if n < 5 then begin F(n + 1); F(n + 2); end end</pre>
---	--

24. Запишите подряд без пробелов и разделителей все числа, которые будут напечатаны на экране при выполнении вызова $F(1)$. Числа должны быть записаны в том же порядке, в котором они выводятся на экран.

Бейсик	Python
<pre>SUB F(n) IF n <= 8 THEN F(n + 3) END IF IF n < 10 THEN PRINT N END IF IF n <= 5 THEN F(n + 4) END IF END SUB</pre>	<pre>def F(n): if n <= 8: F(n + 3) if n < 10: print(n) if n <= 5: F(n + 4)</pre>
Си++	Паскаль
<pre>void F(int n) { if (n <= 8) F(n + 3); if (n < 10) std::cout << n; if (n <= 5) F(n + 4); }</pre>	<pre>procedure F(n: integer); begin if n <= 8 then F(n + 3); if n < 10 then write(n); if n <= 5 then F(n + 4) end; end;</pre>

25. Запишите подряд без пробелов и разделителей все числа, которые будут напечатаны на экране при выполнении вызова $F(5)$. Числа должны быть записаны в том же порядке, в котором они выводятся на экран.

Бейсик	Python
<pre>SUB F(n) PRINT n IF n >= 3 THEN F(n \ 2) F(n - 1) END IF</pre>	<pre>def F(n): print(n, end="") if n >= 3: F(n // 2) F(n - 1)</pre>

END SUB	
Си++	Паскаль
<pre>void F(int n) { std::cout << n; if (n >= 3) { F(n / 2); F(n - 1); } }</pre>	<pre>procedure F(n: integer); begin write(n); if n >= 3 then begin F(n div 2); F(n - 1) end end; end;</pre>

26. Запишите подряд без пробелов и разделителей все числа, которые будут напечатаны на экране при выполнении вызова $F(4)$. Числа должны быть записаны в том же порядке, в котором они выводятся на экран.

Бейсик	Python
<pre>SUB F(n) IF n > 0 THEN F(n - 1) PRINT n F(n - 2) END IF END SUB</pre>	<pre>def F(n): if n > 0: F(n - 1) print(n) F(n - 2)</pre>
Си++	Паскаль
<pre>void F(int n){ if (n > 0){ F(n - 1); std::cout << n; F(n - 2); } }</pre>	<pre>procedure F(n: integer); begin if n > 0 then begin F(n - 1); write(n); F(n - 2) end end; end;</pre>

4.5 Программирование. Задания для самостоятельной подготовки

Три последних задания работы по информатике требуют написания программы на любом **алгоритмическом** языке. Рассматриваются задачи обработки чисел и числовых последовательностей. При оценивании решений учитывается алгоритм, приводя-

щий к корректному результату, знание синтаксиса языка, структура программы (описание переменных, оформление ввода–вывода данных, использование конструкций выбранного языка программирования). Чтобы успешно выполнить задания, необходимо иметь практические навыки написания программ. Для этого предлагается самостоятельно решить следующие задачи.

1. Составить алгоритм вычисления корней квадратного уравнения.
2. Составить алгоритм, в котором выполняется ввод пяти чисел, и после ввода каждого числа выполняет вывод среднего арифметического полученной части последовательности.
3. Составить алгоритм вывода трех чисел в порядке убывания их значений.
4. Заданы три числа: a , b , c . Определить, могут ли они быть сторонами треугольника, и если да, то определить его тип: равносторонний, равнобедренный или разносторонний. Составить алгоритм решения поставленной задачи.
5. Год является високосным, если его номер кратен 4. Из кратных 100 високосными годами являются лишь те года, которые кратны также 400 (например, 1700, 1800, 1900 — не високосные года, 2000 — високосный год). Дано натуральное число N . Определить, является ли високосным год с таким номером.
6. Определить, принадлежит ли точка с координатами X , Y фигуре, заданной левой полуокружностью вертикальной прямой, проходящей через центр окружности. Координаты центра окружности и радиус произвольны (вводятся с клавиатуры).
7. Дано натуральное число N ($N < 9999$).
 - Является ли это число палиндромом (перевертышем) с учетом четырех цифр, как, например числа 2222, 6116, 0440 и т.д.?
 - Верно ли, что число содержит ровно три одинаковые цифры, как, например, числа 6676, 4544, 0006 и т.д.?
 - Верно ли, что все четыре цифры числа различны?
8. Написать программу вычисления факториала заданного числа n .
9. Для данного $N > 7$ найти такие целые неотрицательные a и b , $3 \cdot a + 5 \cdot b = N$.
10. Дано натуральное число.
 - Сколько цифр в заданном числе?
 - Чему равна сумма его цифр?
 - Найти первую цифру числа.
11. Даны натуральное число N , действительные числа $X_1, \dots, X_{3n}$. Вычислить сумму чисел из $X_{n+1}, \dots, X_{3n}$, которые превосходят по величине все числа $X_1, \dots, X_n$.
12. Даны натуральное число N , действительные числа $X_1, \dots, X_n$. Выяснить, является ли последовательность $X_1, \dots, X_n$ упорядоченной

13. Дано натуральное число N и целые числа $A_1, \dots, A_n$. Найти номер первого четного члена последовательности $A_1, \dots, A_n$; если четных членов нет то ответом должно быть число 0.

14. Даны натуральное число N , целые числа $A_1, \dots, A_{30}$, $B_1, \dots, B_{40}$, $C_1, \dots, C_n$. Верно ли, что отрицательный член в последовательности $C_1, \dots, C_n$ встречается раньше, чем в последовательностях $A_1, \dots, A_{30}$ и $B_1, \dots, B_{40}$? Предполагается, что каждая из последовательностей содержит хотя бы один отрицательный член.

15. Дан целочисленный массив из 30 элементов. Элементы массива могут принимать натуральные значения от 1 до 10000 включительно. Опишите на одном из языков программирования алгоритм, который находит минимум среди элементов массива, не делящихся нацело на 6, а затем заменяет каждый элемент, не делящийся нацело на 6, на число, равное найденному минимуму. Гарантируется, что хотя бы один такой элемент в массиве есть. В качестве результата необходимо вывести измененный массив, каждый элемент выводится с новой строки.

16. Дан массив, содержащий 2018 положительных целых чисел, не превышающих 15000. Необходимо уменьшить на удвоенное значение минимального элемента массива те элементы, которые более чем вдвое превышают этот минимум, и вывести измененный массив по одному элементу в строке.

Например, для исходного массива из 5 элементов 40 100 27 90 54 программа должна вывести числа 40 46 27 36 54 по одному числу в строке (минимальный элемент исходного массива равен 27, элементы, большие 54, уменьшены на 54).

Напишите на одном из языков программирования программу для решения этой задачи.

17. Дан целочисленный массив из 30 элементов. Элементы массива могут принимать целые значения от -10000 до 10000 включительно. Опишите на одном из языков программирования алгоритм, который должен находиться на месте многоточия и находить максимальное четное значение среди элементов массива, а затем заменять каждый четный элемент массива числом, равным найденному максимуму.

Гарантируется, что хотя бы одно четное значение среди элементов массива есть. В качестве результата необходимо вывести измененный массив, каждый элемент массива выводить с новой строки.

Например, для массива из шести элементов: 4 -115 7 195 25 -106

программа должна вывести:

```
4
-115
7
195
25
4
```

18. В честь пятилетия открытия магазина «Детский мир» администрация магазина учредила специальный приз, который будет вручен посетителю, дата рождения которого полностью совпадает с датой открытия магазина, если таких посетителей несколько, то специальный приз будет разделен между ними. Если такого посетителя не найдется, то приз, по условиям администрации получит пятый посетитель. Дата открытия магазина — 02.02.2011 г. Необходимо написать программу, эффективную по времени и по используемой памяти, которая определяет порядковый номер призера (или призеров). На вход программы подается число посетителей N ($4 < N < 10000$), далее N дат рождения, записанных в формате ДД ММ ГГ.

Например, для входных данных: 5 02 02 13 03 05 11 02 03 11 22 12 12 12 12 99 программа должна вывести 5.

19. В торговой организации проводится маркетинговое исследование востребованности 11 различных товаров. С этой целью в течение дня фиксируются продажи этих товаров. Необходимо написать эффективную по времени и по используемой памяти программу, которая определяет три самых популярных за день товара.

На вход программы подается число проданных за день товаров $N(3 < N < 10000)$ – количество проданных за день товаров, а далее – N номеров товаров.

Дополнительные условия:

- за день продается больше двух различных товаров;
- общее количество проданных товаров (N) больше трех.

Например, для входных данных: 10 2 3 2 2 4 3 3 4 1 4 программа должна вывести 2 3 4.

20. В организации проводятся наблюдения за автоматическим сигнальным устройством, которое может подавать 10 различных сигналов. С этой целью в течение дня создается массив данных, в котором фиксируются номера сигналов. Необходимо написать программу, эффективную по времени и по используемой памяти, которая определяет три самых редких сигнала. На вход программы подается число поданных за день сигналов $N(3 < N < 10000)$, далее N номеров сигналов. При этом, за день устройство всегда подает больше двух различных сигналов, а общее количество сигналов всегда больше трех.

Например, для входных данных: 10 2 3 5 2 4 3 3 4 1 4 программа должна вывести 1 5 2.

21. По каналу связи передается последовательность положительных целых чисел X , все числа не превышают 1000, их количество N заранее неизвестно. Признаком конца передаваемой последовательности является число 0. Участок последовательности от элемента X_k до элемента X_{k+m} называется подъемом, если на этом участке каждое следующее число больше предыдущего. Высотой подъема называется разность $X_{k+m} - X_k$. Напишите программу, эффективную по памяти и времени, которая вычисляет наибольшую высоту среди всех подъемов последовательности. Если в последовательности нет ни одного подъема, программа выдает 0. Программа должна напечатать результат по следующей форме:

Получено ... чисел Наибольшая высота подъема: ...

Пример входных данных: 144 17 27 3 7 9 11 10 0.

Пример выходных данных: Получено 8 чисел. Наибольшая высота подъема: 10.

22. На вход программы поступает последовательность из N целых положительных чисел, все числа в последовательности различны. Рассматриваются все пары различных элементов последовательности, находящихся на расстоянии не меньше чем 4 (разница в индексах элементов пары должна быть 4 или более, порядок элементов в паре неважен). Необходимо определить количество таких пар, для которых произведение элементов делится на 29.

Описание входных и выходных данных

В первой строке входных данных задаётся количество чисел $N (4 \leq N \leq 1000)$. В каждой из последующих N строк записано одно целое положительное число, не превышающее 10000.

В качестве результата программа должна вывести одно число: количество пар элементов, находящихся в последовательности на расстоянии не меньше чем 4, в которых произведение элементов кратно 29.

Пример входных данных:

7
58
2
3
5
4
1
29

Пример выходных данных для приведённого выше примера входных данных:

5

5 Ответы

5.1 Системы счисления

№ задачи N задания	1	2	3	4	5	6	7	8
1	10100101	11000110	1000011101	1101010001	1111111	11000011	100100001	1000000001
2	53	39	108	93	221	148	458	395
3	456 ₈ 12E ₁₆	407 ₈ 107 ₁₆	713 ₈ 1CB ₁₆	1073 ₈ 23B ₁₆	1313 ₈ 2CB ₁₆	6313 ₈ CCB ₁₆	-	-
4	105 ₈ 45 ₁₆	111 ₈ 49 ₁₆	161 ₈ 71 ₁₆	313 ₈ CB ₁₆	537 ₈ 15F ₁₆	1201 ₈ 281 ₁₆	-	-
5	29	53	141	215	346	489	-	-
6	11 ₁₀	B0 ₁₆ 176 ₁₀	700 ₈ 448 ₁₀	110 ₂ 6 ₁₀	-	-	-	-
7	9 ₁₀	99 ₁₆ 153 ₁₀	55 ₈ 45 ₁₀	101 ₂ 5 ₁₀	-	-	-	-
8	11101110	101001010	100001000	-	-	-	-	-
9	1010101	11101111	10010011	-	-	-	-	-
10	100000011111	100010000111	111010001001	-	-	-	-	-
11	111101	101010	100010	-	-	-	-	-
12	1100×11-100	1100/10-10	1100×10×10	1100-10/10	1100/11-100	-	-	-
13	10	4000	-	-	-	-	-	-
14	23	-	-	-	-	-	-	-
15	3	-	-	-	-	-	-	-
16	+5255 ₈ 4327 ₈ 11604 ₈	- 1536 ₈ 642 ₈ 674 ₈	-	-	-	-	-	-

5.2 Информация и ее кодирование

№	Ответ	№	Ответ
1	16	9	108
2	100	10	192
3	23	11	9375
4	23	12	48
5	22	13	705
6	29	14	1024
7	160	15	376
8	66	16	200

5.3 Основы логики

№	Ответ	№	Ответ
1	3	8	3
2	2	9	4
3	3	10	30
4	1	11	8
5	99	12	10
6	23	13	1
7	19	14	6

5.4 Элементы теории алгоритмов

№	Ответ	№	Ответ
1	105	14	4
2	2349	15	2
3	1009	16	4
4	72	17	10

5	61	18	10
6	35	19	15
7	205	20	6
8	75	21	14
9	499	22	5
10	2226	23	23
11	501	24	7481859
12	27	25	5242312
13	2	26	1231412

6 Список рекомендуемой литературы

1. Ушаков Д.А. Информатика: 20 тренировочных вариантов экзаменационных работ для подготовки к единому государственному экзамену/ Д.М. Ушаков. — АСЕ, 2019. — 287с.
2. ЕГЭ 2018. Информатика. 14 вариантов. Типовые тестовые задания от разработчиков ЕГЭ. В.Р. Лешинер. — М.: Издательство «Экзамен», 2018. — 279с.
3. СДАМ ГИА: РЕШУ ЕГЭ. Образовательный портал для подготовки к экзаменам. Информатика. [Электронный ресурс]: Режим доступа: <https://inf-ege.sdamgia.ru/> (дата обращения: 31.01.2020)