МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«МУРМАНСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Кафедра автоматики
и вычислительной техники

ИНФОРМАТИКА

*Методические указания
к выполнению контрольных работ для обучающихся технических специальностей очной формы обучения по дисциплине «Информатика»*

Мурманск
2019

Составитель: Зоя Алексеевна Бучкова, старший преподаватель кафедры автоматики и вычислительной техники МГТУ.
Рецензент: Николай Николаевич Лейко, кандидат технических наук, доцент кафедры автоматики и вычислительной техники МГТУ.

Методические указания рассмотрены и одобрены кафедрой ______2019 г., протокол №_

ВВЕДЕНИЕ

В данном сборнике методических указаний содержится комплекс заданий для выполнения контрольных работ по информатике, включающий краткие теоретические сведения и варианты заданий.

Методические указания включают две контрольные работы и примеры выполнения заданий по каждому разделу. Это позволяет использовать данное пособие не только для аудиторных занятий, но и для самостоятельной работы обучающихся.

Первая контрольная работа состоит из четырёх заданий, вторая из двух.

Номер варианта контрольных работ следует выбирать согласно последней цифре номера зачётной книжки обучающегося. Если последней цифрой является 0, то номер варианта контрольной работы – 10.

Результаты контрольных работ предоставляются в печатном виде.

ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

Вероятностная оценка количества информации

Количеством информации называют числовую характеристику сигнала, которая не зависит от его формы и содержания и характеризует степень неопределённости, которая исчезает после выбора (получения) сообщения в виде данного сигнала.

В реальной жизни часто приходится иметь дело с явлениями или событиями, точный результат осуществления которых непредсказуем – погода на завтра, цвет шара, вытащенный из урны, сторона монеты, упавшей после подбрасывания и т.д. Число возможных вариантов результата для каждой из ситуаций может быть велико и точно не определено. Идеализированную модель такого рода ситуации называют *опытом*, а возможный вариант результата опыта – *исходом*.

Поскольку число возможных исходов опыта в общем случае велико, то определить точно результат опыта заранее невозможно, т.е. с проведением опыта связана некоторая *неопределённость*. Действительно, в опыте с двумя шарами разного цвета (например, чёрного и белого), находящимися в урне, цвет вынутого шара заранее предсказать нельзя. Ни один из вероятных исходов не имеет преимуществ перед другим. Тогда говорят, что исходы равновероятны.

Определим *вероятность как число, характеризующее меру потенциальной возможности осуществления*.

В опыте с двумя шарами вероятность каждого исхода $p_ч = p_б = 1/2$. Если число шаров разного цвета в урне сделать равным n, то вероятность выбора шара конкретного цвета $p_i = 1/n$.

Впервые количественную оценку неопределённости ввёл в 1928 г. Р. Хартли для опыта X с n различными исходами:

$$H(X) = klog_a n \quad , \qquad (1)$$

где X- информативный параметр сигнала; k-коэффициент пропорциональности; n- число возможных выборов (исходов); a- основание логарифма.

Для получения количественной оценки энтропии обычно используют основание логарифма a, равное двум. Полученная при этом единица измерения количества информации называется *битом*.

Если принять k=1 и опустить a, то формула примет вид:

$$H(X) = logn \quad (2)$$

Однако в оценке Р. Хартли не учтены вероятности различных исходов. К. Шеннон ограничил рамки применимости оценки Р. Хартли случаем, когда все n исходов в опыте X равновероятны (p = 1/n), а затем применил формулу к разновероятным исходам, усреднив полученные неопределённости по всем исходам.

Для опыта $X = \{x_1, x_2, \dots, x_n\}$, где $x_1, x_2, \dots, x_n$ – возможные исходы с вероятностями $p_1, p_2, \dots, p_n$, неопределённость каждого исхода равна $-logp_1, -logp_2, \dots, -logp_n$, а математическое ожидание даёт количественную оценку неопределённости – *энтропию*:

$$H = -\sum_{i=1}^{N} p_i \log_2 p_i \quad (3)$$

При равновероятных выборах все $p_i = 1/N$ и формула (3) преобразуются в формулу Р. Хартли:

$$H = -\sum_{i=1}^{N} p_i \log p_i = -N\frac{1}{N}\log_2\frac{1}{N} = -N\frac{1}{N}(\log_2 1 - \log_2 N) = \log_2 N.$$

Пример 1. В урне находятся 8 белых и 24 чёрных шара. Какое количество информации несёт сообщение о том, что из урны достали белый шар? Чёрный шар?

Решение:

Общее количество шаров в урне 32. Вероятность того, что вытащенный наугад шар - белый, равна 8/32=1/4. Данное сообщение несёт $\log_2(1/(1/4)) = 2$ бита информации.

Для чёрного шара: $log_2(1/(24/32)) = 2 - log_2 3 \approx 0{,}415 < 1$бита информации.

Количество информации тем больше, чем больше свободы в выборе сообщения, т.е. чем более неопределённо или менее вероятно передаваемое сообщение.

Энтропия заранее известного сигнала (значение его информативного параметра априорно известно) равна нулю. Формула для энтропии в этом случае будет состоять из слагаемых только двух видов: либо $1 \cdot log1$ для заранее известного сигнала, либо $0 \cdot log0$, так как вероятность появления всех других вех других равна 0.

Так как $1 \cdot log1 = 0$ и $\lim_{x \to 0}(xlogx) = 0$, то энтропия заранее известного сигнала равна 0.

При двоичном кодировании количество информации на один символ $H = log_2 2 = 1$ бит. Таким образом, количество информации в битах, заключённой в двоичном слове, равно длине самого слова.

Пример 2. Определить количество информации, которую мы получаем в результате бросания несимметричной и симметричной пирамидок.

Решение:

При бросании несимметричной четырёхгранной пирамидки вероятности отдельных событий равны: $p_1 = \frac{1}{2}, p_2 = \frac{1}{4}, p_3 = \frac{1}{8}, p_4 = \frac{1}{8}$.

Количество информации, которую мы получим после бросания несимметричной пирамидки, можно рассчитать по формуле Шеннона:

$$I = -(\frac{1}{2} log_2 \frac{1}{2} + \frac{1}{4} log_2 \frac{1}{4} + \frac{1}{8}\ log_2\ \frac{1}{8} + \frac{1}{8}\ log_2\ \frac{1}{8})\text{ битов}=$$

$$=(\frac{1}{2} log_2\ 2 + \frac{1}{4} log_2 4 + \frac{1}{8} log_2 8 + \frac{1}{8} log_2 8)\text{ битов}=$$

$$=(\frac{1}{2} + \frac{2}{4} + \frac{3}{8} + \frac{3}{8})\text{ битов} = \frac{14}{8}\text{ битов} = 1{,}75\text{ бита.}$$

При бросании симметричной четырёхгранной пирамидки вероятности отдельных событий равны между собой: : $p_1 = p_2 = \ p_3 = p_4 = \frac{1}{4}$.

Количество информации, которую мы получим после бросания симметричной пирамидки, можно рассчитать по формуле: $I = log_2 4 = 2$ бита.

Таким образом, при бросании симметричной пирамидки, когда события равновероятны, мы получаем большее количество информации (2 бита), чем при бросании несимметричной пирамидки, когда события неравновероятны (1,75 бита).

Представление символьных и текстовых данных в двоичном коде

Символы. Для представления символов в числовой форме был предложен метод кодирования, получивший в дальнейшем широкое распространение и для других видов представления нечисловых данных (звуков, изображений и др.). *Кодом* называется уникальное беззнаковое целое двоичное число, поставленное в соответствие некоторому символу. Под *алфавитом* компьютерной системы понимают совокупность вводимых и отображаемых символов. Алфавит компьютерной системы включает в себя арабские цифры, буквы латинского алфавита, знаки препинания, специальные символы и знаки, буквы национального алфавита, символы псевдографики – растры, прямоугольники, одинарные и двойные рамки, стрелки. Первоначально для хранения кода одного символа отвели 1 байт (8 битов), что позволяло закодировать алфавит из 256 различных символов. Система, в которой каждому символу алфавита поставлен в соответствие уникальный код, называется *кодовой таблицей*. Для решения проблемы многообразия кодовых таблиц в 1981 г. Институт стандартизации США принял стандарт кодовой таблицы, получившей название ASCII (American Standard Code of Information Interchange – американский стандартный код информационного обмена). Широкое распространение персональных компьютеров фирмы IBM привело к тому, что стандарт ASCII приобрёл статус международного.

В таблице ASCII содержится 256 символов и их кодов. Таблица состоит из двух частей: основной и расширенной. Основная часть (символы с кодами от 0 до 127 включительно) является базовой, она в соответствии с принятым стандартом не может быть изменена. В неё вошли: управляющие символы (им соответствуют коды с 1 по 31), арабские цифры, буквы латинского алфавита, знаки препинания, специальные символы).

Расширенная часть (символы с кодами от 128 до 255) отдана национальным алфавитам, символам псевдографики и некоторым специальным символам.

В России наиболее распространённой кодовой таблицей алфавита русского языка является «латиница Windows 1251».

В качестве другого примера рассмотрим расширенную таблицу «ГОСТ – альтернативная», на смену которой пришла «латиница Windows 1251».

Во многих странах Азии 256 кодов явно не хватило для кодирования их национальных алфавитов. В 1991 г. Производители программных продуктов и организации, утверждающие стандарты, пришли к соглашению о выработке единого стандарта. Этот стандарт построен по 16 битной схеме кодирования и получил название UNICODE. Он позволяет закодировать $2^{16} = 65536$ символов, которых достаточно для кодирования всех национальных алфавитов в одной таблице. Так как каждый символ этой кодировки занимает два байта (вместо одного, как раньше), все текстовые документы, представленные в UNICODE, стали длиннее в два раза. Современный уровень технических средств нивелирует этот недостаток UNICODE.

Текстовые строки. *Текстовая (символьная) строка* – это конечная последовательность символов. Это может быть осмысленный или произвольный набор, короткое слово или целая книга. Длина символьной строки - это количество символов в ней. Записывается в память символьная строка двумя способами: либо число, обозначающее длину текста, затем текст, либо текст, затем – разделитель строк.

Текстовые документы. Текстовые документы используются для хранения и обмена данными, но сплошной, не разбитый на логические фрагменты текст воспринимается тяжело. *Структурирование* текста достигается *форматированием* – специфическим расположением текста при подготовке его к печати. Для анализа структуры текста были разработаны языки разметки, которые устанавливают *текстовые метки (маркеры или теги*), используемые для обозначения частей документа, записывают вместе с основным текстом в текстовом формате. Программы, анализирующие текст, структурируют его, считывая теги.

Существует формула, которая связывает между собой количество возможных событий N и количество информации I:

$$N=2^I$$

Количество информации в сообщении можно рассчитать по формуле:

$$I_c=I\cdot K,$$

где, I_c – количество информации, которое несёт один символ; K– количество символов в сообщении.

Пример 1. Объём текстовой информации в сообщении на 40 страницах (на странице 40 строк по 80 символов в каждой) в кодировке ASCII равен:

$40\cdot 40\cdot 80=128000$

ASCII – однобайтовая кодировка, 128000·1 байт=128000 байт.

1 Кбайт=1024 байт, следовательно 128000/1024=125 Кбайт.

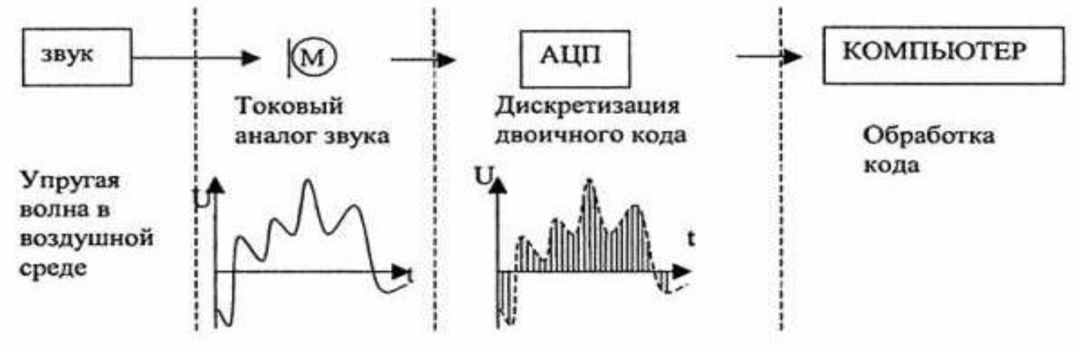
Пример 2. Алфавит некоторого языка состоит из 32 символов. За сколько секунд мы сможем передать текст из 1600 оптимально закодированных символов этого алфавита, если скорость передачи составляет 100 байт в секунду?

Решение:

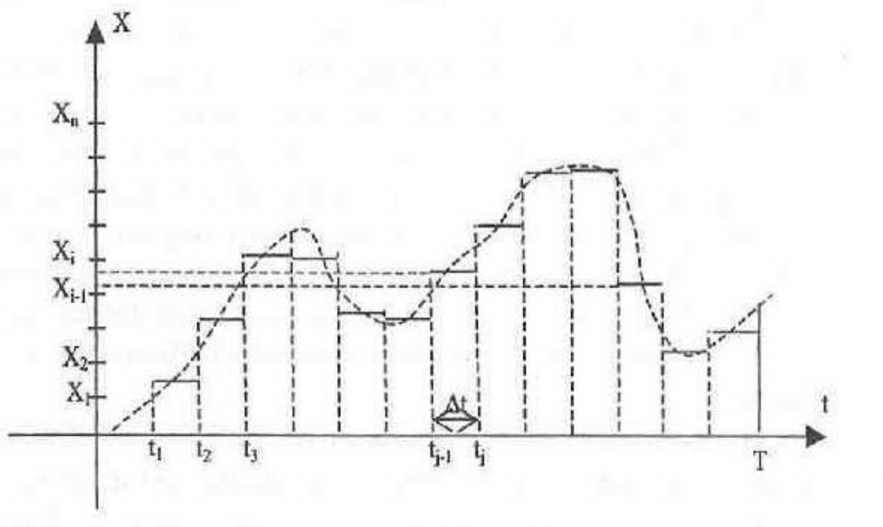
Один символ алфавита можно закодировать 5 битами. Для кодирования 1600 символов потребуется 5·1600 = 8000 бит = 1000 байт. При скорости передачи 100 байт в секунду 1600 символов можно передать за 10 секунд.

Представление звуковых данных в двоичном коде

Звук – это упругая продольная волна в воздушной среде. Чтобы её представить в виде, читаемым компьютером, необходимо выполнить следующие преобразования. Звуковой сигнал преобразовать в электрический аналог звука с помощью микрофона. Электрический аналог получается в непрерывной форме и не пригоден для обработки на цифровом компьютере. Чтобы перевести сигнал в *цифровой код*, надо *пропустить его через аналого-цифровой преобразователь* (АЦП). При воспроизведении происходит обратное преобразование – цифро-аналоговое (ЦАП). АЦП и ЦАП находятся в звуковой карте компьютера.



Во время оцифровки сигнал *дискретизируется* по времени и по уровню.



Дискретизация по времени выполняется следующим образом: весь период времени T разбивается на малые интервалы Δt, точками $t_1, t_2, \ldots, t_n$. Предполагается, что в течении интервала времени Δt уровень сигнала изменяется незначительно и может с некоторым допущением считаться постоянным. Величина $\nu = 1/\Delta t$ называется *частотой дискретизации*. Она измеряется в герцах (Гц) – количество измерений в течении секунды.

Дискретизация по уровню называется *квантованием* и выполняется так: область изменения сигнала от самого малого значения X_{min} до самого большого значения X_{max} разбивается на N равных *кванто*в, промежутков величиной $\Delta X = (X_{min} - X_{max}) / N$. Точками $X_1, X_2, \ldots, X_n$.

$X_i = X_{min} + \Delta X \cdot (i - 1)$.

Каждый квант связывается с его порядковым номером, т.е. целым числом, которое легко может быть преобразовано в двоичной системе счисления. Если сигнал после дискретизации по времени попадает в промежуток $X_{i-1} \leq X \leq X_i$, то ему в соответствие ставится код i.

Если известна глубина кодирования звука, то количество уровней громкостей цифрового звука можно рассчитать по формуле:

$$N=2^i,$$

где N- количество уровней громкости; i – глубина кодирования звука (количество бит).

Объём звукового файла вычисляется по формуле:

$$V=i* \nu*t,$$

где i - глубина звука (количество бит); ν - частота дискретизации (Гц); t- время звучания или записи звука (сек); режим моно=1; режим стерео=2.

Пример 1. Рассчитайте время звучания моноаудиофайла, если при 16-битном кодировании и частоте дискретизации 32 кГц его объём равен 700 Кбайт?

Решение:

700×1024байт/32000[отсчёт/с] / 2[байт/отсчёт]=(716800/32000/2) с =11,2 с.

Пример 2. Оцените информационный объём моноаудиофайла длительностью звучания 1 мин., если глубина кодирования и частота дискретизации звукового сигнала равны соответственно:

а) 16 бит и 8 кГц

б) 16 бит и 24 кГц.

Решение:

а) 16*8000=128000 бит=16000 байт=15,62 Кб – информационный объём файла длительностью 1 секунда
15,62*60=937,5 Кб – информационный объём файла длительностью 1 минута.
б) 16*24000=384000 бит=48000 байт=46,875 Кб – информационный объём файла длительностью 1 секунда
46,875*60=2812,5 Кб=2,8 Мб длительностью 1 минута.

Представление графических данных в двоичном коде

Есть два основных способа представления изображений.

Первый – графические объекты создаются как совокупности линий, векторов, точек – называется *векторной графикой*.

Второй – графические объекты формируются в виде множества точек (пикселей) разных цветов и разных яркостей, распределённых по строкам и столбцам, - называется *растровой графикой*.

Модель RGB. Чтобы оцифровать цвет, его необходимо измерить. Немецкий учёный Грасман сформулировал три закона смешения цветов:

1) Закон *трёхмерности* – любой цвет может быть представлен комбинацией трёх основных цветов;
2) Закон *непрерывности* – к любому цвету можно подобрать бесконечно близкий;
3) Закон *аддитивности* – цвет смеси зависит только от цвета составляющих.

За основные три цвета приняты красный (Red), зелёный (Green), синий (Blue). В модели RGB любой цвет получается в результате сложения основных цветов. Каждый составляющий цвет характеризуется своей яркостью, поэтому модель называется аддитивной. Эта схема применяется для создания графических образов в устройствах излучающих свет, - мониторах, телевизорах.

Модель CMYK. В полиграфических системах напечатанный на бумаге графический объект сам не излучает световых волн. Изображение формулируется на основе отражённой волны от окрашенных поверхностей. Окрашенные поверхности, на которые падает белый свет (т.е. сумма всех цветов), должны поглотить (т.е. вычесть) все составляющие цвета, кроме того, цвет которой мы видим. Цвет поверхности можно получить красителями, которые поглощают, а не излучают. Например, если мы видим зелёное дерево, то это означает, что из падающего белого цвета, т.е. суммы красного, зелёного, синего, поглощены красный и синий, а зелёный отражён. Цвета красителей должны быть дополняющими:

голубой (Cyan=B+G), дополняющий красного;

пурпурный (Magenta=R+B), дополняющий зелёного;

жёлтый (Yellow=R+G), дополняющий синего.

Но так как цветные красители по отражающим свойствам не одинаковы, то для повышения контрастности применяется чёрный цвет (black). Модель CMYK названа по первым буквам слов Cyan, Magenta, Yellow и последней букве слова black. Так как цвета вычитаются, то модель называется *субстрактивной*.

Оцифровка изображения. При оцифровке изображение с помощью объектива проецируется на светочувствительную матрицу m строк и n столбцов, называемую растром. Каждый элемент матрицы – мельчайшая точка, при цветном изображении состоящая из трёх светочувствительных (т.е. регистрирующих яркость) датчиков красного, зелёного, жёлтого цвета. Далее оцифровывается яркость каждой точки по каждому цвету последовательно по всем строкам растра.

Если для кодирования яркости каждой точки использовать по одному байту (8 бит) на каждый из трёх цветов (всего 3· =24 бита), то система обеспечит представление 2^{24}≈16,7 млн распознаваемых цветов, что близко цветовосприятию человеческого зрения. Режим представления цветной графики двоичным кодом из 24 разрядов называется *полноцветным* или True Color. Например, скромный по современным меркам экран монитора имеет растр 800×600 точек, изображение, представленное в режиме True Color, займёт 800×600×3=1440000 байт.

В случае, когда не требуется высокое качество отображения цвета, применяют режим High Color , который кодирует одну точку растра двумя байтами (16 разрядов дают 2^{16} ≈ 65,5 тысячи цветов).

Режим, который при кодировании одной точки растра использует один байт, называется *индексным*, в нём различаются 256 цветов. Этого недостаточно, чтобы передать весь диапазон цветов. Код каждой точки при этом выражает собственно не цвет, а некоторый номер цвета (индекс) из таблицы цветов, называемой *палитрой*. Палитра должна прикладываться к файлам с графическими данными и используется при воспроизведении изображения.

Объём графического файла определяется по формуле:

$$V=a*b*i,$$

где i - глубина цвета (количество бит); a, b - разрешение изображения по горизонтали и вертикали.

Пример 1. Растровое изображение размером 64×64 пикселя занимает 4 Килобайта памяти. Чему равно максимально возможное количество цветов в палитре изображения?

Решение:

Узнаем, сколько битовых разрядов используется для кодирования одного пикселя. Всего пикселей 64×64=4096.

4 Килобайта = 4096 байтов. Получается, что на кодирование цвета каждого пикселя отводится 1 байт памяти, т.е. 8 битов. Далее используем формулу, связывающую количество двоичных разрядов (У) для кодирования цвета с количеством цветов (N):

$$N=2^Y$$

$N=2^8=256$ (каждый пиксель может иметь один цвет из 256).

Пример 2. Сколько секунд потребуется модему, передающему сообщения со скоростью 28800 бит/сек, чтобы передать цветное растровое изображение размером 640×480 пикселей, при условии, что цвет каждого пикселя кодируется тремя байтами?

Решение:

Данное изображение имеет следующий объём в битах: $640\times480\times3\times8=$ $=2^{13}\times9\times100$. При указанной скорости такое изображение будет передаваться $2^{13} : 2^5=2^8=256$ сек.

Системы счисления

Система счисления – совокупность приёмов и правил наименования и обозначения чисел, позволяющих установить взаимно однозначное соответствие между любым числом и его представлением в виде конечного числа символов.

Все системы счисления можно разделить на позиционные и непозиционные.

Непозиционная система счисления – система, в которой символы, обозначающее то или иное количество, не меняют своего значения в зависимости от местоположения в изображении числа.

Запись числа A в непозиционной системе счисления D может быть представлена выражением: $A_D = D_1 + D_2 + \ldots + D_N = \sum_{i=1}^{N} D_i$, где A_D- запись числа A в системе счисления D; D_i- символы системы.

К *непозиционной системе счисления* относится, например, римская, символы алфавита которой и обозначаемое ими количество представлены в таблице.

Римские цифры	I	V	X	L	C	D	M
Значение (обозначаемое количество)	1	5	10	50	100	500	1000

Запись чисел в этой системе счисления осуществляется по следующим правилам:

1) если цифра слева меньше, чем цифра справа, то левая цифра вычитается из правой (IV: 1<5, следовательно, 5 – 1 = 4, XL: 10<50, следовательно, 50 – 10 = 40);
2) если цифра справа меньше или равна цифре слева, то эти цифры складываются (VI: 5+1 = 6, VIII: 5+1+1+1 = 8, XX: 10+10=20).

Пример 1. Число 1964 в римской системе счисления имеет вид MCMLXIV (M – 1000, CM – 900, LX – 60, IV – 4).

Систему счисления, в которой значение цифры определяется её местоположением в изображении числа, называют *позиционной*.

Возможно бесчисленное множество позиционных систем счисления: двоичная, троичная, четверичная и т.д. Запись чисел в каждой из систем счисления с основанием p означает сокращённую запись выражения:

$$A(p) = a_n p^n + a_{n-1} p^{n-1} + \ldots + a_1 p^1 + a_0 p^0 + a_{-1} p^{-1} + \ldots + a_{-m} p^{-m} = \sum_{k=-m}^{n} a_k p^k,$$

где a_i- цифры системы счисления; n и m - число целых и дробных разрядов, соответственно; A_p- запись числа A в p-ичной системе счисления.

Алфавиты некоторых систем счисления:

Основание	Система счисления	Алфавит системы счисления
2	Двоичная	0,1
3	Троичная	0,1,2
4	Четверичная	0,1,2,3
5	Пятеричная	0,1,2,3,4
8	Восьмеричная	0,1,2,3,4,5,6,7
10	Десятичная	0,1,2,3,4,5,6,7,8,9
12	Двенадцатеричная	0,1,2,3,4,5,6,7,8,9,A,B
16	Шестнадцатеричная	0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F

Пример 2. Перевести данное число $A_{10} = 47$ из десятичной системы счисления в двоичную:

$47 : 2 = 23(1)$;

$23 : 2 = 11(1)$;

$11 : 2 = 5(1)$;

$5 : 2 = 2(1)$;

$2 : 2 = 1(0)$;

$1 : 2 = 0(1)$.

$A_2 = 101111$.

Пример 3. Перевести данное число в десятичную систему счисления.

а) $1000001_{(2)}$.

$1000001_{(2)}=1 \times 2^6+0\times 2^5+0\times 2^4+0\times 2^3+0 \times 2^2+ 0\times 2^1+1\times 2^0 = 64+1=65_{(10)}$.

Замечание. Очевидно, что если в каком-либо разряде стоит нуль, то соответствующее слагаемое можно опускать.

б) $1000011111,0101_{(2)}$.

$1000011111,0101_{(2)}=1\times 2^9 + 1\times 2^4 + 1\times 2^3 + 1\times 2^2 + 1\times 2^1 + 1\times 2^0 + 1\times 2^{-2} + 1\times 2^{-4} = 512 + 16 + 8 + 4 + 2 + 1 + 0,25 + 0,0625 = 543,3125_{(10)}$.

в) $1216,04_{(8)}$.

$1216,04_{(8)}=1\times 8^3+2\times 8^2+1\times 8^1+6\times 8^0+4\times 8^{-2} = 512+128+8+6+0,0625 = 654,0625_{(10)}$.

г) $29A,5_{(16)}$.

$29A,5_{(16)} = 2\times 16^2+9\times 16^1+10\times 16^0+5\times 16^{-1} = 512+144+10+0,3125 = 656,3125_{(10)}$.

Пример 4. Перевести число $0,2_{10}$ в двоичную систему счисления:

$0,2 \cdot 2 = 0,4 = 0 + 0,4 \Rightarrow b_{-1} = 0$;

$0{,}4 \cdot 2 = 0{,}8 = 0 + 0{,}8 \Rightarrow b_{-2} = 0$;

$0{,}8 \cdot 2 = 1{,}6 = 1 + 0{,}6 \Rightarrow b_{-3} = 1$;

$0{,}6 \cdot 2 = 1{,}2 = 1 + 0{,}2 \Rightarrow b_{-4} = 1$ и. т. д.

$A_2 = 0.(0011)$ – периодическая дробь.

Пример 5. Перевести число $0{,}36_{10}$ в восьмеричную систему счисления:

$0{,}36 \cdot 8 = 2{,}88 = 2 + 0{,}88 \Rightarrow b_{-1} = 2$;

$0{,}88 \cdot 8 = 7{,}08 = 7 + 0{,}08 \Rightarrow b_{-2} = 7$;

$0{,}08 \cdot 8 = 0{,}64 = 0 + 0{,}64 \Rightarrow b_{-3} = 0$ и. т. д.

$A_8 = 0{,}270$.

Пример 6. Перевести число $67532{,}107_8$ в двоичную систему счисления.

Заменим каждую цифру трёхзначным двоичным числом:

6 7 5 3 2 1 0 7
| | | | | | | |
110 111 101 011 010, 001 000 111

т.е. $67532{,}107_8 = 110111101011010{,}001000111$

Пример 7. Перевести число 10111011101,1101 в восьмеричную систему счисления.

Заменим каждую триаду восьмеричным числом:

010 111 011 101, 110 100
| | | | | |
2 7 3 5 6 4

т.е. 10111011101,1101 = 2735,64.

Пример 8. Перевести число $35B{,}451E_{16}$ в двоичную систему счисления.

Заменим каждую шестнадцатеричную цифру двоичной тетрадой:

3 5 B, 4 5 1 F
| | | | | | |
0011 0101 1011, 0100 0101 0001 1111

т.е. $35B{,}451E_{16} = 1101011011{,}0100010100011111$.

Пример 9. Сложить числа:

а)$10001101{,}1_{(2)}+111011{,}11_{(2)}=11001001{,}01_{(2)}$.

б)$17_{(8)}+6_{(8)}=25_{(8)}$.

в) $3B3{,}6_{(16)} + 38B{,}4_{(16)} = 73E{,}A_{(16)}$.

```
 10001101,1          17          3B3,6
 +  111011,11      +   6       +38B,4
-------------------       ---        ---------
 11001001,01         25          73E,A
```

Пример 10. Выполните вычитание:

а)$11001001{,}01_{(2)}-111011{,}11_{(2)}=10001101{,}10_{(2)}$.

б)$311{,}2_{(8)}-73{,}6_{(8)}=215{,}4_{(8)}$.

в) $C9{,}4_{(16)} – 3B{,}6_{(16)} = 8D{,}8_{(16)}$.

```
  11001001,01            311,2        C9,4
 - 00111011,11         -    73,6      -3B,6
 ---------------------       --- ------       ----------
   10001101,10            215,4        8D,8
```

Пример 11. Выполнить умножение:

а)$100111_{(2)}\times1000111_{(2)}=101011010001_{(2)}$.

б)$1170{,}64_{(8)}\times46{,}3_{(8)}=57334{,}134_{(8)}$.

в) $61{,}A_{(16)} \times 40{,}D_{(16)} = 18B7{,}52_{(16)}$.

```
        100111            1170,64           61,A
       *1000111            *   46,3          *40,D
     -------------        --------------      ----------
        100111            355 234           4F 52
   +     100111        +    7324 70         +  1868
       100111            47432 0           ----------
     100111              -------------        18B7,52
   -------------------    57334,134
     101011010001
```

Пример 12. Выполнить деление:

а) $1011011101001_{(2)} : 1110011_{(2)} = 110011$.

```
101101110100   |1110011
-1110011          110011
  1000100
-1110011
10101100
- 1110011
   1110011
-  1110011
```

0

Основы машинной арифметики

В цифровых автоматах применяются три формы записи (кодирования) целых чисел со знаком: прямой, обратный и дополнительный коды.

Прямой код двоичного числа включает в себя код знака (знак " + " соответствует 0, знак " - " - 1) и абсолютное значение этого числа:

X_{10} = -13, X_2 = -1101, [X_2]пр = 1|1101,

X_{10} = 9, X_2 = 1001, [X_2]обр = 0|1001.

Вертикальная линия отделяет знаковый разряд от абсолютного значения числа.

Обратный код положительных чисел такой же, как и прямой код, в знаковом разряде - 0. Для получения обратного кода отрицательного числа его разряды инвертируются (0 заменяется на 1 и, наоборот), в знаковом разряде - цифра 1:

X_{10} = 12, X_2 = 1100, [X_2]пр = [X_2]обр = 0|1100,

X_{10} = -7, X_2 = -0111, [X_2]обр = 1|1000.

Дополнительный код положительных чисел такой же, как и прямой код. У отрицательных чисел дополнительный код равен результату суммирования обратного кода числа с единицей младшего разряда: X_{10} = 18, X_2 = 10010, [X_2]пр = [X_2]обр = [X_2]доп = 0|10011,

X_{10}= -11, X_2 = -1011, [X_2]доп = [X_2]обр+1 = 1|0100+1 = 1|0101.

Использование чисел со знаком (прямого кода представления чисел) усложняет структуру ЭВМ. В этом случае операция сложения двух чисел, имеющих разные знаки, должна быть заменена на операцию вычитания меньшей величины из большей и присвоения результату знака большей величины. Поэтому в современных ЭВМ, как правило, отрицательные числа представляют в виде дополнительного или обратного кода, что при суммировании двух чисел с разными знаками позволяет заменить вычитание на обычное сложение и упростить тем самым конструкцию арифметико – логического устройства.

Пример 1. Сложение обратных кодов [доп. 1].

1) X и У положительные.

3 0’ 0000011
+7 +0’ 0000111
10 0’ 0001010

2) X положительное, У отрицательное и по абсолютной величине больше, чем X.

+3 0’ 0000011
-10 +1’ 1110101 обратный код числа -10
-7 1’ 1111000 обратный код числа -7

Получен корректный результат в обратном коде. При переводе в прямой код биты цифровой части результата инвертируются: 1’ 0000111 = -7_{10}.

3) X положительное, У отрицательное и по абсолютной величине меньше, чем X.

+10 +0’ 0001010
-3 1’ 1111100 обратный код числа -3
7 0’ 0000110
+1
0’ 0000111

Компьютер исправляет полученный первоначально некорректный результат (6 вместо 7) *переносом единицы* из знакового разряда в младший разряд суммы.

4) X и У отрицательные.

-3 +1’ 1111100 обратный код числа -3
+-7 1’ 1111000 обратный код числа -7
-10 1’ 1110100
+1
1’ 1110101 обратный код числа -10

При переводе результата в прямой код биты цифровой части числа инвертируются:
1’ 0001010 = -10_{10}.

Логические основы ПК

Алгебра логики – раздел математики, изучающий высказывания, рассматриваемые со стороны их логических значений (истинности или ложности) и логических операций над ними.

Высказывание – некоторое предложение, в отношении которого можно однозначно сказать, истинно оно или ложно.

Высказывания, которым приписаны численные значения 1 или 0, называются *логическими переменными*.

В алгебре логики определены следующие логические операции:

¬, ¯ - логическое отрицание (“НЕ”, инверсия);

*, &, ^, AND – логическое умножение (“И”, конъюнкция);

+, v, OR – логическое сложение (“ИЛИ”, дизъюнкция);

⊕, ∀ - логическое исключающее “ИЛИ” (исключающая дизъюнкция);

→ - логическое следование (импликация);

↔, = - эквивалентность (двойная импликация);

| -функция Шеффера;

↓ - стрелка Пирса, или функция Вебба º;

$\overline{\rightarrow}$ *и* $\overline{\leftarrow}$ - обратная импликация, или коимпликация правая и левая;

~ - эквивалентность или функция тождества.

Значение каждой логической функции описывается общей таблицей истинности. *Таблица истинности* представляет собой таблицу, устанавливающую соответствие между возможными значениями наборов переменных и значениями функции.

Отрицание – унарная (т.е. для одного операнда) логическая операция. Ей соответствует языковая конструкция, использующая частицу НЕ. Это правило можно записать в виде следующей таблицы:

A	$\overline{A}$
0	1
1	0

Конъюнкцией (логическим умножением) двух высказываний **A** и **B** является новое высказывание **C**, которое истинно только тогда, когда истинны оба высказывания, записывается **C=A∧B** или **C=A&B** .

A	**B**	**A&B**
0	0	0
0	1	0
1	0	0
1	1	1

Дизъюнкцией (логическим сложением) двух высказываний **A** и **B** является новое высказывание **C**, которое истинно, если истинно хотя бы одно высказывание. Записывается **C=A∨B**.

A	**B**	**A∨B**
0	0	0
0	1	1
1	0	1
1	1	1

Импликацией двух высказываний **A** *(*называется *посылкой)* и **B** (называется *заключением)* является новое высказывание **C,** которое ложно только тогда, когда посылка истинна, а заключение ложно, записывается **C=A→B.**

A	**B**	**A→B**
0	0	1
0	1	1
1	0	0
1	1	1

Импликация имеет следующие свойства:

A→B ≠ B→A

A→A=1

0→A=1

1→A=A

A→1=1

A→0=$\overline{\mathrm{A}}$

Эквиваленцией двух высказываний **A** и **B** является новое высказывание **C**, которое истинно только тогда, когда оба высказывания имеют одинаковые значения истинности, записывается **C=A↔B**.

A	**B**	**A↔B**
0	0	1
0	1	0
1	0	0
1	1	1

Эквиваленция имеет следующие свойства:

A↔B = B↔A

A↔B = $\overline{\mathrm{B}}\leftrightarrow\overline{\mathrm{A}}$

A↔1=A

A↔0 = $\overline{\mathrm{A}}$

Чтобы избежать большого количества скобок в булевских функциях, принято следующее соглашение о старшинстве операций.

Первыми выполняются операции в скобках, затем операции в следующем порядке: отрицание, конъюнкция и дизъюнкция слева направо, импликация, эквиваленция.

Теоремы алгебры логики:

1. $\overline{\overline{A}} \equiv A$ - закон двойного отрицания
2. $A\&B \equiv B\&A$ - коммутативный закон для конъюнкции
3. $A \vee B \equiv B \vee A$ - коммутативный закон для дизъюнкции
4. $(A\&B)\&C \equiv A\&(B\&C)$ - ассоциативный закон для конъюнкции
5. $(A \vee B) \vee C \equiv A \vee (B \vee C)$ - ассоциативный закон для дизъюнкции
6. $A\&(B \vee C) \equiv (A\&B) \vee (A\&C)$ - дистрибутивный закон
7. $A \vee (B\&C) \equiv (A \vee B)\&(A \vee C)$ - дистрибутивный закон
8. $A\&A \equiv A$ - закон идемпотентности для конъюнкции
9. $A \vee A \equiv A$ - закон идемпотентности для дизъюнкции
10. $A \vee \overline{A} = 1$ закон исключения третьего
11. $A \& \overline{A} = 0$ закон противоречия
12. $(A\&B) \vee A \equiv A$ - закон поглощения
13. $(A \vee B)\&A \equiv A$ - закон поглощения
14. $\overline{A\&B} \equiv \overline{A} \vee \overline{B}$- закон де Моргана
15. $\overline{A \vee B} \equiv \overline{A}\&\overline{B}$- закон де Моргана
16. $A\&1 \equiv A$ - закон единицы для конъюнкции
17. $A\&0 \equiv 0$- закон нуля для конъюнкции
18. $A \vee 1 \equiv 1$ - закон единицы для дизъюнкции
19. $A \vee 0 \equiv A$ - закон нуля для дизъюнкции
20. $A \rightarrow B = \overline{A} \vee B$
21. $A \leftrightarrow B = (A \rightarrow B) \& (B \rightarrow A) = (\overline{A} \vee B) \& (A \vee \overline{B}) = (A \& B) \vee (\overline{A} \& \overline{B})$
22. $A \& (\overline{A} \vee B) = A \& B$
23. $A \vee (\overline{A} \& B) = A \vee B$

Существует несколько стандартных форм, к которым приводятся логические выражения с помощью эквивалентных преобразований (формул 1-23).

Первая из них – *дизъюнктивная нормальная форма* (ДНФ), имеет вид $\mathbf{A1 \vee A2 \vee \ldots \vee A_n}$, где каждое из составляющих высказываний есть конъюнкция простых высказываний и их отрицаний, например:

$$\mathbf{B = (\overline{A1} \& A2 \& A3) \vee (A4 \& A5)}.$$

Вторая из них – *конъюнктивная нормальная форма (КНФ),* имеет вид **$A1 \wedge A2 \wedge \ldots \wedge A_n$**, где каждое из составляющих высказываний есть дизъюнкция простых высказываний и их отрицаний, например:

$$\mathbf{B=(\overline{A1} \vee A2 \vee A3) \& (A4 \vee A5) \& A6}.$$

Пример 1. Построить таблицу истинности для функции

$$F(x1, x2, x3) = (x1 \to x2) \to \overline{x3}$$

Вначале выписываются значения, которые могут принимать набор переменных в этой функции. В общем случае, если переменных n, то различных n-мерных наборов переменных существует 2^n. Затем вычисляется значение функции на каждом наборе. Любая, рассматриваемая логическая функция представляет собой суперпозицию элементарных логических функций и может быть вычислена последовательно при помощи подстановок определённых ранее значений.

Итак, определим значение наборов переменных. Их восемь:

$x1\ x2\ x3$	$x1 \to x2$	$\overline{x3}$	$(x1 \to x2) \to \overline{x3}$
0 0 0	1	1	1
0 0 1	1	0	0
0 1 0	1	1	1
0 1 1	1	0	0
1 0 0	0	1	1
1 0 1	0	0	1
1 1 0	1	1	1
1 1 1	1	0	0

Пример 2. Доказать тождество

$$A \cdot B + \overline{A} \cdot B = B,$$

$$(A + B) \cdot (\overline{A} + B) = B$$

Доказательство первого тождества проводится с использованием первого закона дистрибутивности:

$$A \cdot B + \overline{A} \cdot B = B \cdot (A + \overline{A}) = B \cdot 1 = B$$

Доказательство второго тождества проводится с использованием второго закона дистрибутивности: $(A + B) \cdot (\overline{A} + B) = A \cdot \overline{A} + B = B$.

Пример 3. Доказать закон свёртки логического выражения [доп. 4]:

$$A \cdot B + \overline{A} \cdot C + B \cdot C = A \cdot B + \overline{A} \cdot C.$$

Данное тождество можно доказать, последовательно используя законы работы с логическими константами, дистрибутивности, идемпотентности и склеивания:

$$A\cdot B+\overline{A}\cdot C+B\cdot C=A\cdot B\cdot(C+\overline{C})+\overline{A}\cdot C\cdot(B+\overline{B})+B\cdot C\cdot(A+\overline{A})=$$
$$=A\cdot B\cdot C+A\cdot B\cdot\overline{C}+\overline{A}\cdot B\cdot C+\overline{A}\cdot\overline{B}\cdot C+A\cdot B\cdot C+\overline{A}\cdot B\cdot C=$$
$$=(A\cdot B\cdot C+A\cdot B\cdot\overline{C})+(\overline{A}\cdot B\cdot C+\overline{A}\cdot\overline{B})+(\overline{A}\cdot\overline{B}+\overline{A}\cdot B\cdot C)=$$
$$=A\cdot B+\overline{A}\cdot C+\overline{A}\cdot C=A\cdot B+\overline{A}\cdot C$$

Пример 4. Используя основные равносильности алгебры логики, а также равносильности $A\to B\equiv\overline{A}\vee B$, упростить формулу $(X\to Y)\to Y$.

Решение:

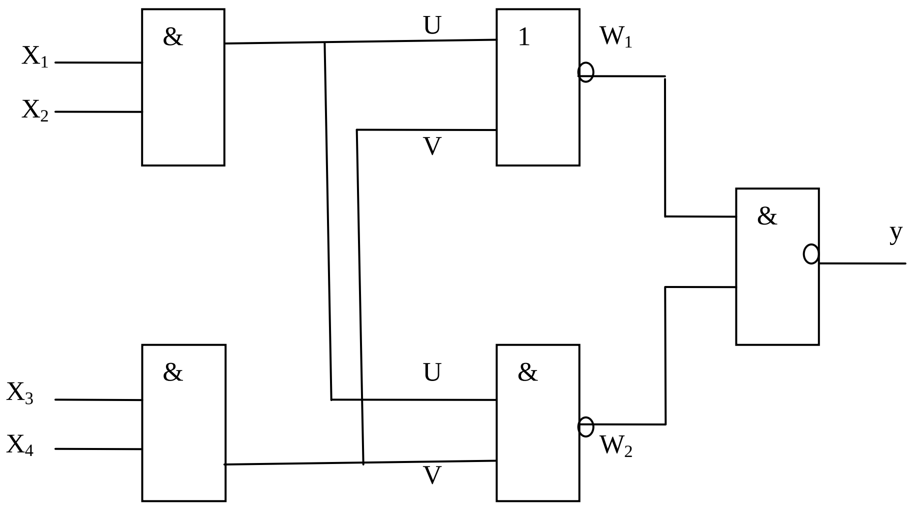
$$(X\to Y)\to Y\equiv(\overline{X}\vee Y)\to Y\equiv\overline{\overline{X}\vee Y}\vee Y\equiv(\overline{X}\&\overline{Y})\vee Y\equiv(X\&\overline{Y})\vee Y\equiv$$
$$\equiv(X\vee Y)\&(\overline{Y}\vee Y)\equiv(X\vee Y)\&1\equiv X\vee Y$$

Пример 5. Для заданной комбинационной схемы постройте аналитическое выражение, упростите его с помощью равносильных преобразований и, если возможно, нарисуйте упрощённую схему. Здесь $U=x1\&x2$, $V=x3\&x4$

$W_1=\overline{U\vee V}=\overline{(x1\&x2)\vee(x3\&x4)}$,

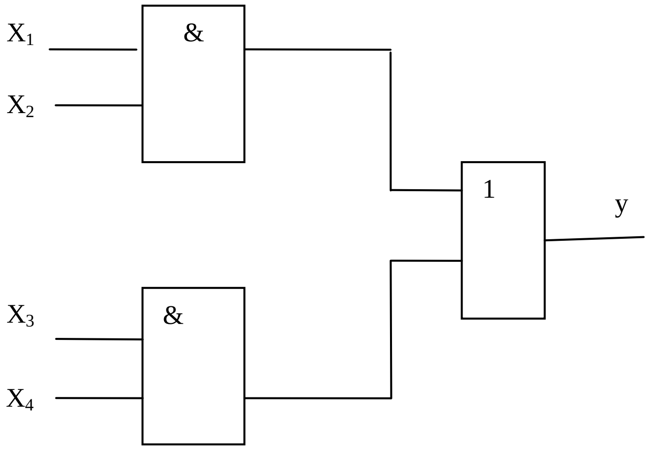
$W_2=\overline{U\&V}=\overline{(x1\&x2)\&(x3\&x4)}$,

$y=\overline{W_1\&W_2}=\overline{\overline{(x1\&x2)\vee(x3\&x4)}\&\overline{(x1\&x2)\&(x3\&x4)}}$.



Преобразуем последнее выражение по закону де Моргана. Получаем $y=((x1\&x2)\vee(x3\&x4))\vee(x1\&x2)\&(x3\&x4)$.

Используя законы ассоциативности и правила приоритета логических операций, получаем $y = x1x2 \vee x3x4 \vee x1x2x3x4$. Далее используем правило поглощения $A \vee AB \equiv A$, в результате получим упрощённую формулу, равносильную данной $y = x1x2 \vee x3x4$. Ей соответствует упрощённая комбинационная схема:



КОНТРОЛЬНАЯ РАБОТА №1

Задание №1

Вариант 1

В конкурсе участвовали 20 студентов, 8 школьников и 4 учащихся колледжа. Какое количество информации несёт сообщении о том, что победил школьник, считая, что победа любого из участников равновероятна?

Вариант 2

В русском языке буква «М» встречается с вероятностью 0,025, а буква «А» - с вероятностью 0,06. Какое слово более информативно: «МАМА» или «МММ»?

Вариант 3

В озере живут караси и окуни. Подсчитано, что карасей 1500, а окуней - 500. Сколько информации содержится в сообщениях о том, что рыбак поймал карася, окуня, поймал рыбу?

Вариант 4

В корзине лежат 32 красных и чёрных шара. Среди них 4 красных шара. Сколько информации несёт сообщение, что достали шар любого цвета?

Вариант5

В русском языке буква «М» встречается с вероятностью 0,025, а буква «У» - с вероятностью 0,02. Определите количество информации в слове «МУМУ».

Вариант 6

В непрозрачном мешочке хранятся 10 белых, 20 красных, 30 синих и 40 зелёных шариков. Какое количество информации будет содержать зрительное сообщение о цвете вынутого шарика?

Вариант 7

В коробке 32 карандаша, все карандаши разного цвета. Наугад вытащили красный. Какое количество информации при этом было получено?

Вариант 8

В библиотеке 16 стеллажей с книгами, по 8 полок в каждом. Нужный учебник находится на второй полке четвёртого стеллажа. Какое количество информации мы получим, что нашлась нужная книга?

Вариант 9

В некоторой игре одновременно подбрасывают монету и игральный кубик. Сколько информации несёт сообщение о результате падения этих двух предметов?

Вариант 10

Для ремонта использовали белую, синюю и жёлтую краски. Израсходовали одинаковое количество белой и синей краски. Сообщение о том, что закончилась банка белой краски, несёт 2 бита информации. Синей краски израсходовали 8 банок. Сколько банок жёлтой краски израсходовали на ремонт?

Задание №2

Вариант 1

Растровое изображение размером 128×128 пикселей занимает 4 Килобайта памяти. Чему равно максимальное количество цветов, используемых в изображении?

Вариант 2

Текст занимает полных 5 страниц. На каждой странице размещается 30 строк по 70 символов в строке. Какой объём оперативной памяти (в байтах) займёт этот текст?

Вариант 3

Количество цветов, воспроизводимых на экране сотового телефона, равно 1024, разрешение экрана 128×64. Чему равен минимальный объём видеопамяти (Кбайт)?

Вариант 4

Растровый графический файл содержит чёрно-белое изображение с 2 градациями цвета (чёрный и белый) размером 800×600 точек. Определите необходимый для кодирования цвета точек размер этого файла на диске в байтах.

Вариант 5

Определите, какой объём памяти необходим для цифрового хранения 50-минутной серии чёрно-белого фильма «Семнадцать мгновений весны», если телевизионные кадры сменяют друг друга 25 раз в секунду, а один телевизионный кадр состоит из 625 строк по 880 точек, каждая из которых кодируется одним из 256 оттенков серого цвета.

Вариант 6

Объём свободной памяти на диске – 5,25 Мб, разрядность звуковой платы - 16. Какова длительность звучания цифрового аудиофайла, записанного с частотой дискретизации 22,05 кГц?

Вариант 7

Оцените информационный объём цифрового стереозвукового файла длительностью звучания 1 секунда при среднем качестве звука (16 битов, 24000 измерений в секунду).

Вариант 8

Две минуты записи цифрового аудиофайла занимают на диске 5,1 Мб, частота дискретизации 22050 Гц. Какова разрядность аудиоадаптера?

Вариант 9

Объём сообщения, содержащего 2048 символов, составил 1/512 часть Мбайта. Определите мощность алфавита.

Вариант 10

Определите количество уровней звукового сигнала при использовании устаревших 8-битных звуковых карт.

Задание №3

Вариант 1

1. Сложить числа: а)$10001101{,}1_{(2)}+111011{,}11_{(2)}$ б)$17_{(8)}+6_{(8)}$ в) $3B3{,}6_{(16)} + 38B{,}4_{(16)}$	2. Выполните вычитание: а)$11001001{,}01_{(2)}-111011{,}11_{(2)}$ б)$311{,}2_{(8)}-73{,}6_{(8)}$ в) $C9{,}4_{(16)} – 3B{,}6_{(16)}$
3. Перевести числа в 10-ю систему счисления: а)$10010011111_{(2)}$ б)$1372{,}12_{(8)}$ в) $3CA{,}7D_{(16)}$	4. Перевести 10-тичные числа в 2-ю, 8-ю 16-ю системы счисления: а)$1802_{(10)}$ б)$286{,}06_{(10)}$ в) $622_{(10)}$

Вариант 2

1. Сложить числа: а) $100101{,}101_{2)}$ $+11101{,}11_{2}$ б)$17_{(8)}+7_{(8)}$ в) $3B3{,}6_{(16)} + 38B{,}4_{(16)}$	2. Выполните вычитание: а) $100101{,}101_{(2)}$- $11101{,}11_{(2)}$ б)$311{,}2_{(8)}$-$72{,}6_{(8)}$ в) $C9{,}4_{(16)} – 3B{,}6_{(16)}$
3. Перевести числа в 10-ю систему счисления: а)$11100101010{,}011_{(2)}$ б)$2136{,}31_{(8)}$ в)$1C3{,}A2_{(16)}$	4. Перевести 10-тичные числа в 2-ю, 8-ю 16-ю системы счисления: а)$1731_{(10)}$ б)$476{,}91_{(10)}$ в) $622_{(10)}$

Вариант 3

1. Сложить числа: а) $1100000{,}101_{(2)}$ $+1111{,}111_{(2)}$ б)$18_{(8)}+6_{(8)}$ в) $3B3{,}6_{(16)} + 38B{,}4_{(16)}$	2. Выполните вычитание: а) $1100000{,}101_{(2)}$ - $1111{,}111_{(2)}$ б)$312{,}2_{(8)}$-$73{,}6_{(8)}$ в) $C9{,}4_{(16)} – 3B{,}6_{(16)}$
3. Перевести числа в 10-ю систему счисления: а)$11100101010{,}011_{(2)}$ б)$2136{,}31_{(8)}$ в)$1C3{,}A2_{(16)}$	4. Перевести 10-тичные числа в 2-ю, 8-ю 16-ю системы счисления: а)$1731_{(10)}$ б)$476{,}91_{(10)}$ в) $522_{(10)}$

Вариант 4

1. Сложить числа: а) $1011101{,}101_{(2)}$ $+10111{,}011_{(2)}$ б)$17_{(8)}+8_{(8)}$ в) $3B3{,}6_{(16)} + 38B{,}4_{(16)}$	2. Выполните вычитание: а) $1011101{,}101_{(2)}$ $-10111{,}011_{(2)}$ б)$311{,}2_{(8)}$-$71{,}6_{(8)}$ в) $C9{,}4_{(16)} – 3B{,}6_{(16)}$

3. Перевести числа в 10-ю систему счисления: а)$10100101111,101_{(2)}$ б)$2451,23_{(8)}$ в)$2BA,D3_{16}$	4. Перевести 10-тичные числа в 2-ю, 8-ю 16-ю системы счисления: а)$1630_{(10)}$ б)$609,11_{(10)}$ в) $422_{(10)}$

Вариант 5

1. Сложить числа: а) $1010111,101_{(2)} + 11100,111_{(2)}$ б)$19_{(8)}+6_{(8)}$ в) $3B3,6_{(16)} + 38B,4_{(16)}$	2. Выполните вычитание: а) $1011101,101_{(2)} - 10111,011_{(2)}$ б)$311,2_{(8)}-73,5_{(8)}$ в) $C9,4_{(16)} - 3B,6_{(16)}$
3. Перевести числа в 10-ю систему счисления: а)$11100101101,1011_{(2)}$ б)$1275,46_{(8)}$ в)$23A,E7_{(16)}$	4. Перевести 10-тичные числа в 2-ю, 8-ю 16-ю системы счисления: а)$1856_{(10)}$ б)$552,5_{(10)}$ в) $322_{(10)}$

Вариант 6

1. Сложить числа: а)$10001101,1_{(2)}+111011,11_{(2)}$ б)$17_{(8)}+6_{(8)}$ в) $3B3,6_{(16)} + 38B,4_{(16)}$	2. Выполните вычитание: а)$11001001,01_{(2)}-111011,11_{(2)}$ б)$311,2_{(8)}-73,6_{(8)}$ в) $C9,4_{(16)} - 3B,6_{(16)}$
3. Перевести числа в 10-ю систему счисления: а)$10010011111_{(2)}$ б)$1372,12_{(8)}$ в) $3CA,7D_{(16)}$	4. Перевести 10-тичные числа в 2-ю, 8-ю 16-ю системы счисления: а)$1802_{(10)}$ б)$286,06_{(10)}$ в) $622_{(10)}$

Вариант 7

1. Сложить числа: а) $100101,101_{2)} + 11101,11_{2}$ б)$17_{(8)}+7_{(8)}$ в) $3B3,6_{(16)} + 38B,4_{(16)}$	2. Выполните вычитание: а) $100101,101_{(2)} - 11101,11_{(2)}$ б)$311,2_{(8)}-72,6_{(8)}$ в) $C9,4_{(16)} - 3B,6_{(16)}$

3. Перевести числа в 10-ю систему счисления: а)$11100101010,011_{(2)}$ б)$2136,31_{(8)}$ в)$1C3,A2_{(16)}$	4. Перевести 10-тичные числа в 2-ю, 8-ю 16-ю системы счисления: а)$1731_{(10)}$ б)$476,91_{(10)}$ в) $622_{(10)}$

Вариант 8

1. Сложить числа: а) $1100000,101_{(2)}$ $+1111,111_{(2)}$ б)$18_{(8)}+6_{(8)}$ в) $3B3,6_{(16)}$ + $38B,4_{(16)}$	2. Выполните вычитание: а) $1100000,101_{(2)}$ - $1111,111_{(2)}$ б)$312,2_{(8)}-73,6_{(8)}$ в) $C9,4_{(16)}$ – $3B,6_{(16)}$
3. Перевести числа в 10-ю систему счисления: а)$11100101010,011_{(2)}$ б)$2136,31_{(8)}$ в)$1C3,A2_{(16)}$	4. Перевести 10-тичные числа в 2-ю, 8-ю 16-ю системы счисления: а)$1731_{(10)}$ б)$476,91_{(10)}$ в) $522_{(10)}$

Вариант 9

1. Сложить числа: а) $1011101,101_{(2)}$ $+10111,011_{(2)}$ б)$17_{(8)}+8_{(8)}$ в) $3B3,6_{(16)}$ + $38B,4_{(16)}$	2. Выполните вычитание: а) $1011101,101_{(2)}$ $-10111,011_{(2)}$ б)$311,2_{(8)}-71,6_{(8)}$ в) $C9,4_{(16)}$ – $3B,6_{(16)}$
3. Перевести числа в 10-ю систему счисления: а)$10100101111,101_{(2)}$ б)$2451,23_{(8)}$ в)$2BA,D3_{16}$	4. Перевести 10-тичные числа в 2-ю, 8-ю 16-ю системы счисления: а)$1630_{(10)}$ б)$609,11_{(10)}$ в) $422_{(10)}$

Вариант 10

1. Сложить числа: а) $1010111{,}101_{(2)}$ $+11100{,}111_{(2)}$ б)$19_{(8)}+6_{(8)}$ в) $3В3{,}6_{(16)}$ + $38В{,}4_{(16)}$	2. Выполните вычитание: а) $1011101{,}101_{(2)}$ $-10111{,}011_{(2)}$ б)$311{,}2_{(8)}-73{,}5_{(8)}$ в) $С9{,}4_{(16)}$ – $3В{,}6_{(16)}$
3. Перевести числа в 10-ю систему счисления: а)$11100101101{,}1011_{(2)}$ б)$1275{,}46_{(8)}$ в)$23А{,}Е7_{(16)}$	4. Перевести 10-тичные числа в 2-ю, 8-ю 16-ю системы счисления: а)$1856_{(10)}$ б)$552{,}5_{(10)}$ в) $322_{(10)}$

Задание №4

Запишите числа Х и У в прямом, обратном и дополнительном кодах. Выполните сложение чисел в обратном и дополнительном кодах. Результат переведите в прямой код. Полученный результат проверьте, используя правила двоичной арифметики:

Вариант	**Число Х**	**Число У**
1	Х = - 100101	У = 11101
2	Х = - 110101	У = 11101
3	Х = -1000111	У = 11101
4	Х = -1010001	У = 10011
5	Х = -1101001	У = 10111
6	Х = -101001	У = 10111
7	Х = - 110101	У = 10011
8	Х = -1000111	У = 11101
9	Х = - 100101	У = 11101
10	Х = - 110101	У = 10011

КОНТРОЛЬНАЯРАБОТА №2

Задание №1

Для заданного логического выражения:

a) Построить таблицу истинности;
b) Упростить высказывания, используя законы алгебры логики;

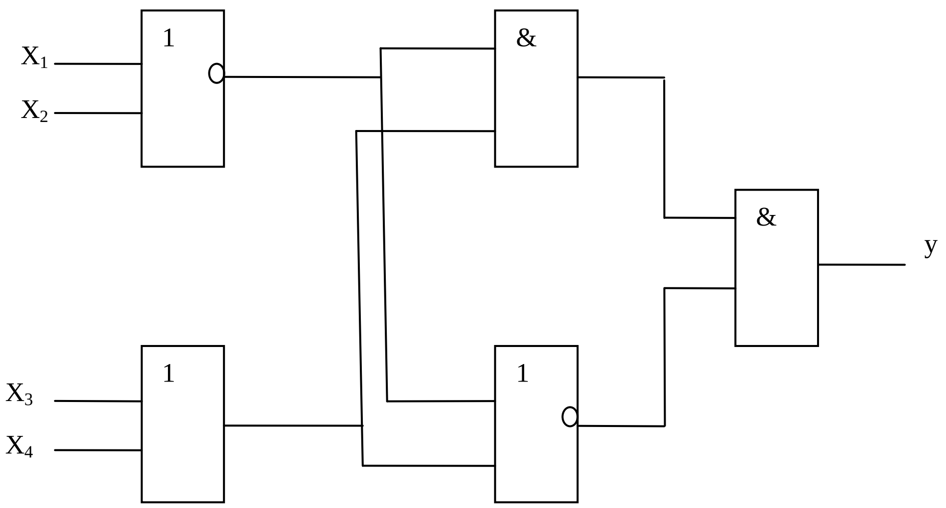
c) Полученный результат проверить, построив для него таблицу истинности.

1	$(A \leftrightarrow B) \vee A\overline{B} \vee C$
2	$(AC \to B) \vee A\overline{C}$
3	$A\overline{B} \vee (A \leftrightarrow C)B$
4	$(\overline{A} \leftrightarrow B)(A \to BC)$
5	$(B \to C) \vee (B \to AC)$
6	$(AB \to C) \vee A\overline{C}$
7	$(AC \to \overline{B}) \vee B\overline{C}$
8	$\overline{(A \to B)}(C\overline{A} \to \overline{B})$
9	$(BC \to A) \vee A\overline{C}$
10	$B \vee (A \leftrightarrow CB) \vee A\overline{C}$

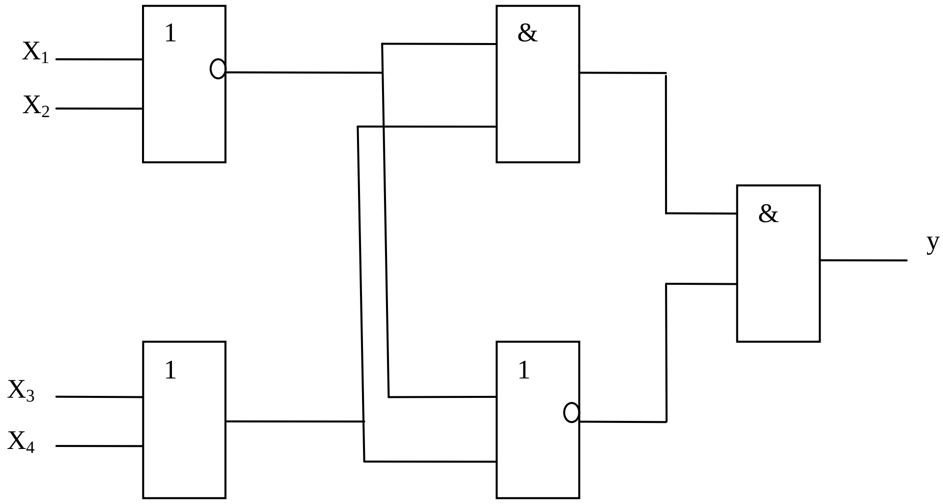
Задание №2

Для заданной комбинационной схемы постройте аналитическое выражение, упростите его с помощью равносильных преобразований и, если возможно, нарисуйте упрощённую схему.

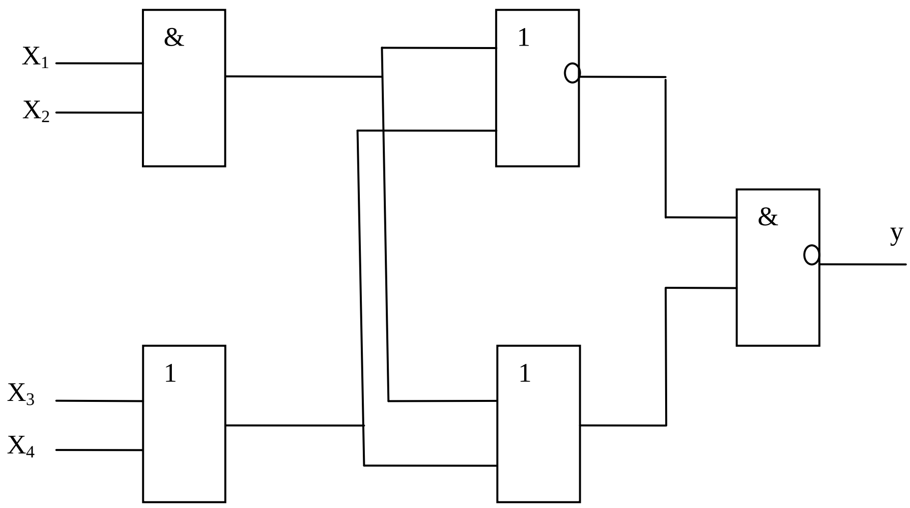
Вариант 1



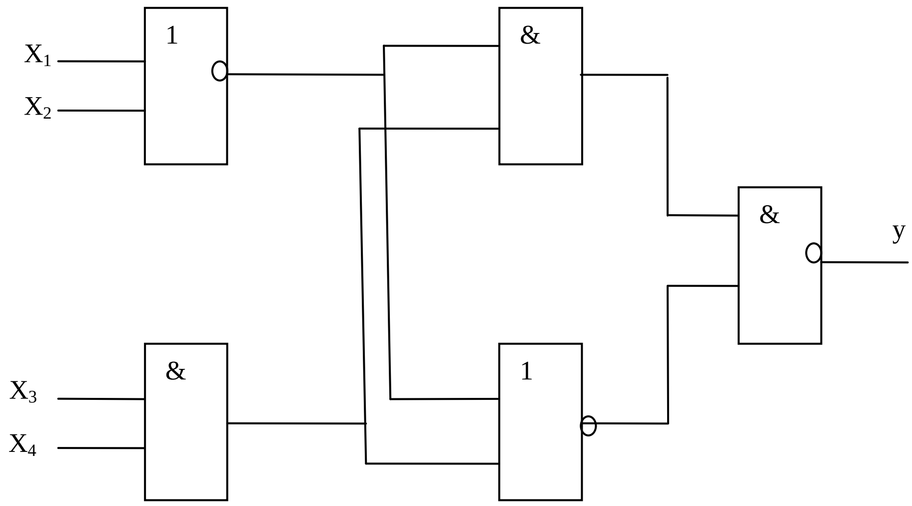
Вариант2



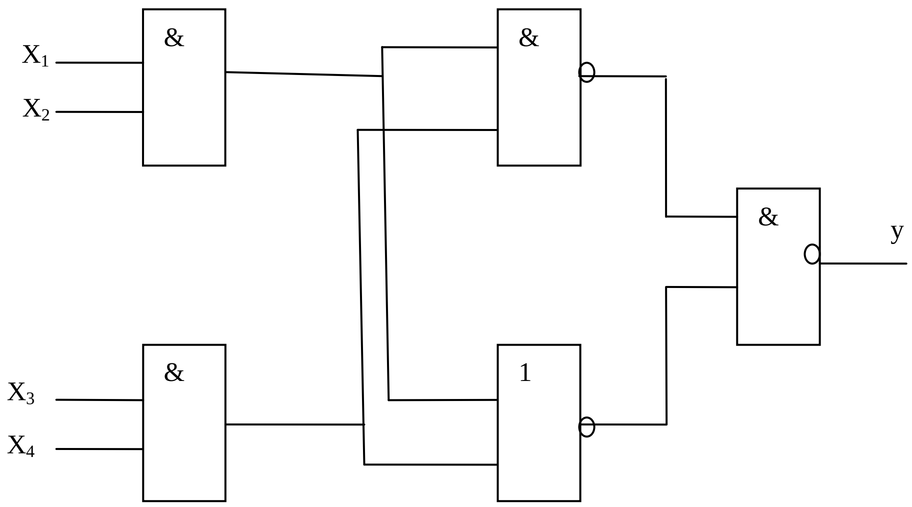
Вариант 3



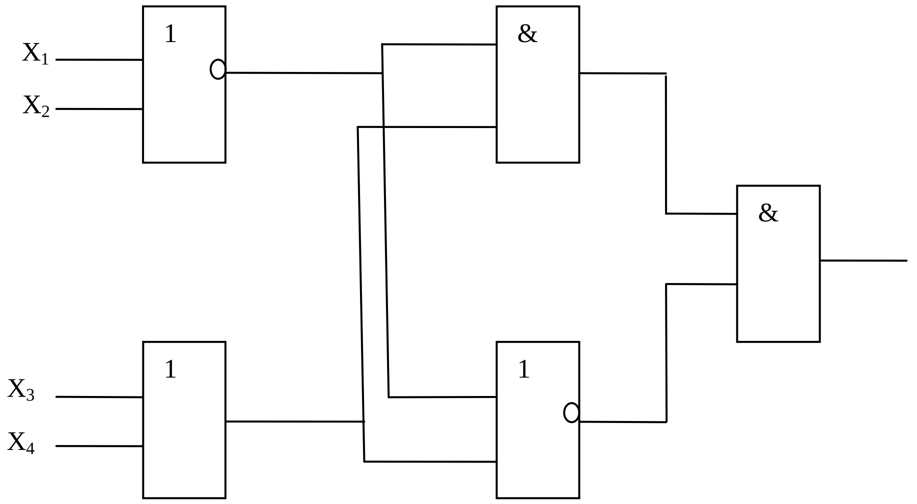
Вариант 4



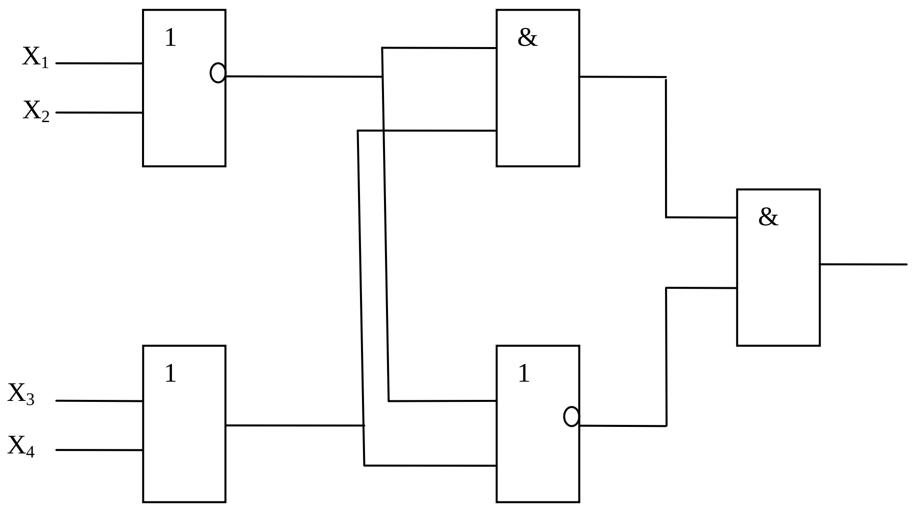
Вариант 5



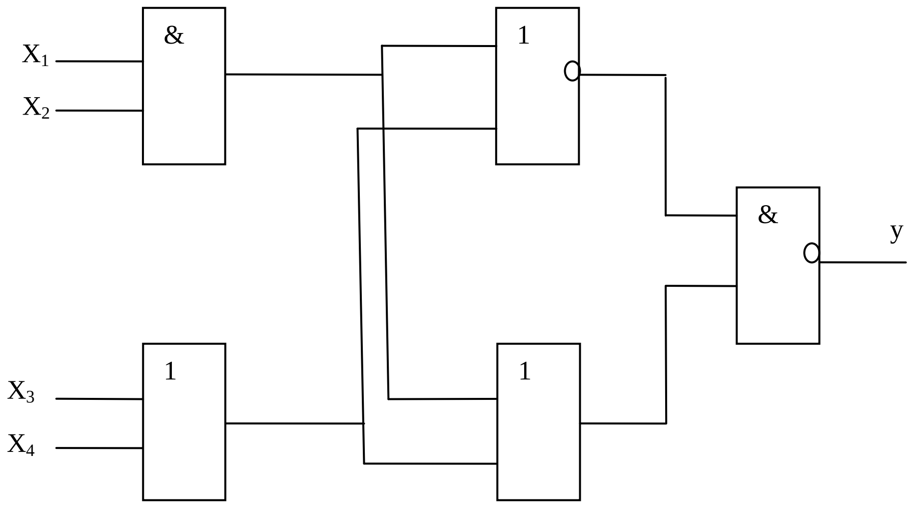
Вариант 6



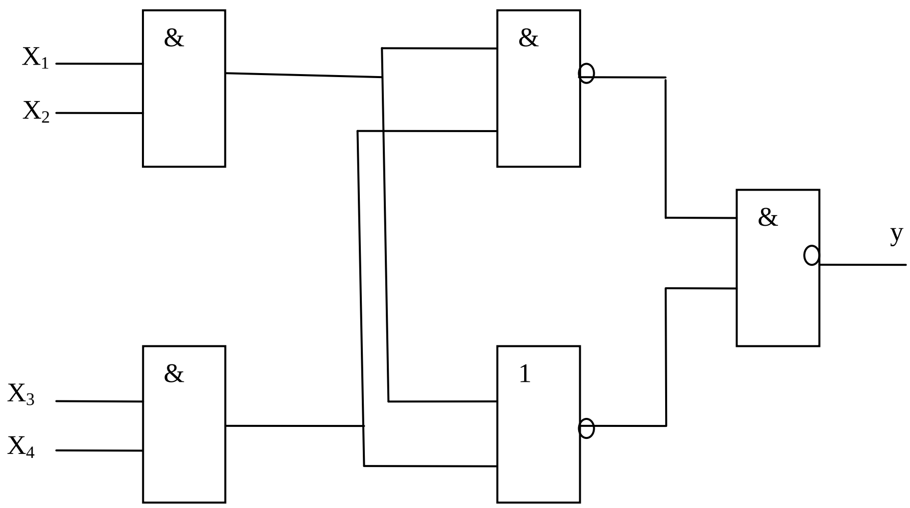
Вариант 7



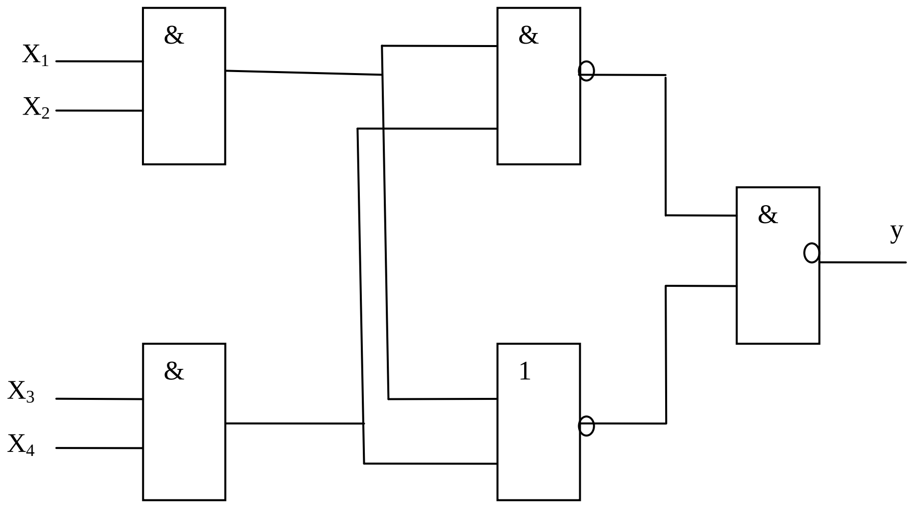
Вариант 8



Вариант 9



Вариант 10


&
&
&
1
&
X1
X2
X3
X4
y

СПИСОК ЛИТЕРАТУРЫ

Основная литература:

1. Макарова Н.В., Волков В. Б. Информатика: учебник для вузов. - Санкт-Петербург [и др.] : Питер, 2012. - 573 с.: ил. - (Учебник для вузов) (Стандарт третьего поколения). Количество – 44.
2. Симонович С.В. Информатика. Базовый курс: учебник для вузов. – 3-е изд. - Санкт-Петербург [и др.]: Питер, 2012. - 637 с.: ил. - (Учебник для втузов) (Стандарт третьего поколения). Количество -50.
3. Угринович Н.Д. Информатика. 10 класс. Базовый уровень:– М.: Бином. Лаборатория знаний, 2016. – 288 с.: ил.

Дополнительная литература:

1. Акулов О.А. Информатика: базовый курс : учеб. для студентов вузов, бакалавров, магистров, обучающихся по направлению «Информатика и вычисл. техника» / О.А. Акулов, Н. В. Медведев. – 5-е изд., испр. и доп. – М.: Омега Л, 2008. – 574 с.
2. Иопа Н.И. Информатика (для технических направлений): учебное пособие – 2-е изд., стер. – М.: КНОРУС, 2012. – 472 с. – (Бакалавриат).
3. Соболь Б.В. Информатика: учебник. - Изд. 3-е, дополн. и перераб. - Ростов н/Д: Феникс, 2007. - 446 [1] с. -(Высшее образование).
4. Соболь Б.В. Практикум по информатике: учебное пособие. –Ростов н/Д: Феникс, 2009. – 509 [1] с. – (Высшее образование).