Ассоциация научно-технических организаций "Уральский профессиональный форум"
Автономная некоммерческая профессиональная образовательная организация "Современный цифровой колледж при Западно-уральском институте экономики и права" (АНПОО "СЦК при ЗУИЭП")

УТВЕРЖДАЮ
Директор
_____________ /И.И. Лобанова/
«___» февраля 2024 г.

ОЦЕНОЧНЫЕ СРЕДСТВА ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ
В ФОРМЕ ДИФФЕРЕНЦИРОВАННОГО ЗАЧЕТА
по дисциплине
ЕН.02 ДИСКРЕТНАЯ МАТЕМАТИКА С ЭЛЕМЕНТАМИ МАТЕМАТИЧЕСКОЙ ЛОГИКИ
по специальности
09.02.07 Информационные системы и программирование
квалификация «Программист»

форма обучения: очно-заочная

Вводится с 01.09.2024

Пермь 2024

РАССМОТРЕНО
на заседании Педагогического совета
протокол от «26» февраля 2024 г. № 4

1. Общие положения

Контрольно-оценочные средства (КОС) являются составной частью образовательной программы среднего профессионального образования по подготовке специалистов среднего звена **09.02.07 Информационные системы и программирование, квалификация «Программист»,** и предназначены для контроля и оценки образовательных достижений обучающихся, освоивших программу учебной дисциплины **«Дискретная математика».**

КОС включают контрольные материалы для проведения промежуточной аттестации в форме дифференцированного зачета.

КОС разработаны на основании:

Положения о Фонде оценочных средств (ФОС);

Рекомендаций по разработке контрольно-оценочных средств (КОС);

Рабочей программы учебной дисциплины.

2. Результаты освоения дисциплины, подлежащие проверке

Код	Умения	Знания
ОК 01. ОК 02. ОК 04. ОК 05. ОК 09. ОК 10. ПК 1.1.-1.6 ПК 11.1.-11.6 ПК 2.1.-ПК 2.5. ПК 4.1.-ПК 4.4.	Применять логические операции, формулы логики, законы алгебры логики. Формулировать задачи логического характера и применять средства математической логики для их решения.	Основные принципы математической логики, теории множеств и теории алгоритмов. Формулы алгебры высказываний. Методы минимизации алгебраических преобразований. Основы языка и алгебры предикатов. Основные принципы теории множеств.

4. Содержание КОС

Содержание банка КОС в полной мере отражает требования ФГОС по специальности и содержания рабочей программы учебной дисциплины. В состав банка включены теоретические вопросы и практические задания.

4.1. Теоретические задания (ТЗ):

1. Взаимосвязь дискретной математики с другими дисциплинами. Практические проблемы, изучаемые методами дискретной математики

2. Составные высказывания. Простейшие связки. Логические отношения.

3. Варианты импликации.

4. Основные законы, определяющие свойства логических операций.

5. Булевы функции.

6. Свойства элементарных булевых функций.

7. Дизъюнктивные и конъюнктивные нормальные формы алгебры высказываний.

8. Совершенная дизъюнктивная и совершенная конъюнктивная нормальные формы.

9. Многочлены Жегалкина.

10. Специальные классы булевых функций: функции, сохраняющие единицу, функции, сохраняющие нуль, самодвойственные функции, линейные функции, монотонные функции. Теорема Поста о функциональной полноте.

11. Понятие множества. Способы задания множества. Подмножества. Операции над множествами.

12. Соотношения между множествами и составными высказываниями.

13. Соотношение между высказываниями и соответствующими им множествами истинности.

14. Абстрактные законы операций над множествами.

15. Кортежи и декартово произведение множеств. Степень множества.

16. Бинарные отношения в множестве и их свойства.

17. Отношения строгого и нестрогого порядка.

18. Отображение множеств. Функции.

19. Определенность и неопределенность функций. Композиция отображений.

20. Метод математической индукции. База индукции. Индукционный переход. Полная и неполная индукция.

21. Основные правила комбинаторики. Методы алгоритмического перечисления (генерации) основных комбинаторных объектов: перестановка, сочетание, размещение.

22. Комбинация элементов с повторениями. Бином Ньютона.

23. Предикаты. Применение предикатов в алгебре.

24. Булева алгебра предикатов.

25. Кванторы. Формулы логики предикатов.

26. Равносильные формулы логики предикатов. Приведенные и нормальные формы в логике предикатов.

27. Исчисления предикатов.

28. Основные понятия теории графов. Степень вершины. Маршрут, цепи, циклы. Связность графа.

29. Ориентированные графы.

30. Изоморфизм графов.

31. Плоские графы. Операции над графами.

32. Способы задания графов. Некоторые типы графов.

33. Сети. Сетевые модели представления информации. Применение графов и сетей.

34. Вычислимые функции и алгоритмы.

35. Рекурсивные функции.

36. Нормальный алгоритм Маркова.

37. Машины Тьюринга.

38. Понятия конечного автомата. Определения и способы задания конечного автомата.

39. Примеры конечных автоматов.

40. Канонические уравнения автомата.

4.2. Практические задания (ПЗ):

1. Докажите тождественную истинность формулы $\overline{X}$ **→(X→У).**

2. С помощью таблиц истинности проверить, являются ли эквивалентными высказывания: **f_1= X^ (Y→Z) и f_2= ($\overline{X}$^Y) ˅ (X^Z).**

3. Определите для каждого из следующих высказываний, будет ли оно логически истинным, противоречивым: ни тем, ни другим.

а) X↔X, б)X↔$\overline{X}$, в)(X˅У)↔(X^У), г)(X→$\overline{У}$)→(У→$\overline{X}$), д)(X→Y)^(Y→Z)^$\overline{(X \to Z)}$, е)(X→У)→X, ж)((X→У)→X

4. Доказать закон отрицания конъюнкции ($\overline{A \wedge B} \leftrightarrow \overline{A} \vee \overline{B}$)
5. Найти значение $\overline{A \wedge B} \leftrightarrow \overline{A} \vee \overline{B}$ и убедиться, что при всех значениях А и В - это истинное значение.
6. С помощью основных равносильностей доказать закон обобщенного склеивания $x\,y \vee \bar{x}\,z = x\,y \vee \bar{x}\,z \vee y\,z$.

7. Составьте таблицу истинности булевой функции трех переменных $f(x_1, x_2, x_3)$ = $x_1 \oplus x_2 \to \overline{x3} \vee x_1 \,|(\overline{x2} \wedge \overline{x1})$ и найдите ее двоичный набор.

8. Докажите эквивалентность функции: $f_{(x,\,y,\,z)}$ = x^(x˅z)^(y˅z) и $f_{(x,\,y,\,z)}$ = (x^y)˅(x^z).

9. Найдите СДНФ и СКНФ функции **$f_{(x,\,y,\,z)}$,** заданной следующей таблицей истинности:

x_1	x_2	x_3	$f(x_1, x_2, x_3)$
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

10. Опрос 100 студентов выявил следующие данные о числе студентов, изучающих различные иностранные языки: английский – 28; немецкий - 30; французский – 42, английский и немецкий – 8; английский и французский – 10; немецкий и французский – 5; все три языка – 3.

1) Сколько студентов не изучают ни одного иностранного языка?

2) Сколько студентов изучают один французский язык?

3) Сколько студентов изучают немецкий язык в том и только в том случае, если они изучают французский язык?

Решение. Нарисовать диаграмму Эйлера-Венна в виде трех кругов, обозначающих множество студентов, изучающих соответственно французский, немецкий и английский языки. В каждую из 8-ми областей вписать данные, используя приведенные цифры. Начинать с конца списка и двигаться к началу.

Ответ: 1) 20; 2) 30; 3) 38.

11. Изобразите с помощью диаграмм Эйлера-Венна множества:
1) $A \subset B$ и $B \subset C$; 2) $A \subset B$; $B \subset C$ и $A \setminus B = \emptyset$; 3) $A \subset B$; $B \subset C$ и $C = A \cup B$;

4) $A \subset B$; $B \subset C$ и $A \cap B \neq \emptyset$; 5) $(A \setminus B) \cap (B \setminus A)$.

12. Воспользовавшись диаграммой Эйлера-Венна, определите, какие из следующих высказываний истинны: **а)** $\mathbf{X} \vee \overline{\boldsymbol{X}}$; **б)** $\mathbf{X} \wedge \overline{\boldsymbol{X}}$; **в)** $\mathbf{X} \vee (\overline{\boldsymbol{X}} \wedge \boldsymbol{Y})$;
г) $\mathbf{X} \to (\mathbf{Y} \to \mathbf{X})$; **д)** $\mathbf{X} \wedge \overline{(\boldsymbol{Y} \to \boldsymbol{X})}$.

13. Пусть A = {1, 2}. Выписать все элементы декартова произведения $A \times A$.

14. Рассмотрим два множества A = {a, b, c, d, e, f, g, h} и B = {1, 2, 3, 4, 5, 6, 7, 8}.

15. Проверьте на линейность функцию $f(x_1, x_2, x_3)$, если ее двоичный набор F= 11100001.

16. Найдите правую и левую области отношения R = {‹1, 5› ; ‹1, 6› ; ‹1, 7›}.

17. Если A = {2, 3,4, 5, 6, 7, 8}, запишите бинарное отношение R = {‹x, y› : x, y $\in$ A, x делит y, и x ≤ 3}.

18. Являются ли следующие отношения функциями:

1) $\{\langle 1, 2\rangle ; \langle 2, 3\rangle ; \langle 3, 2\rangle\}$; 2) $\{\langle 1, 2\rangle ; \langle 1, 3\rangle ; \langle 2, 3\rangle\}$;
3) $\{x, x^2 - 2x - 3 : x \in R\}$?

19. Футбольный мяч сшит из 32 лоскутов: белых шестиугольников и чёрных пятиугольников. Каждый чёрный лоскут граничит только с белыми, а каждый белый – с тремя чёрными и тремя белыми. Сколько лоскутов белого цвета?

20. Из 4 первокурсников, 5 второкурсников и 6 третьекурсников надо выбрать 3 студента на конференцию. Сколькими способами можно осуществить этот выбор, если среди выбранных должны быть студенты разных курсов?

21. Сколько можно составить четырехзначных чисел так, чтобы любые две соседние цифры были различны?

22. Сколькими способами можно рассадить 5 человек за круглым столом (рассматривается только расположение сидящих относительно друг друга)?

23. Сколькими способами можно распределить 15 выпускников по 3 районам, если в одном из них имеется 8, в другом - 5 и в третьем – 2 вакантных места?

24. Известно, что 7 студентов сдали экзамен по дискретной математике на хорошо и отлично. Сколькими способами могли быть поставлены им оценки?

25. Группа студентов изучает 10 различных дисциплин. Сколькими способами можно составить расписание занятий в понедельник, если в этот день должно быть 4 разных занятия?

26. Из 60 вопросов, входящих в экзаменационные билеты, студент знает 50. Найти вероятность того, что среди трех наугад выбранных вопросов студент знает: а) все вопросы, б) два вопроса.

27. Во взводе три сержанта и 30 солдат. Сколькими способами можно выделить одного сержанта и трех солдат для патрулирования?

28. В барабане револьвера 7 гнезд, из них в 5 заложены патроны. Барабан приводится во вращение, потом нажимается спусковой курок. Какова вероятность того, что, повторив такой опыт 2 раза подряд: а) револьвер оба раза не выстрелит, б) оба раза револьвер выстрелит.

29. Решить уравнение: $5C_x^3 = C_{x+2}^4$

30. Решить уравнение: $C_{x-3}^2 = 21$

31. Решить уравнение: $A_x^3 = \frac{1}{20} A_x^4$

32. Решить уравнение: $C_x^3 + C_x^2 = 15(x-1)$

33. Решить уравнение: $C_x^3 = \frac{5x\,(x-3)}{4}$

34. Записать предложение: «прямая а параллельна прямой b» с помощью предиката.

35. Записать с помощью предиката: «Аксиома: через две различные точки проходит единственная прямая» (Если две точки принадлежат двум прямым, то эти прямые совпадают).

36. Между планетами введено космическое сообщение по следующим маршрутам: З-К, П-В, З-П, П-К, К-В, У-М, М-С, С-Ю, Ю-М, М-У. Можно ли добраться с З до М?

37. Аркадий, Борис. Владимир, Григорий и Дмитрий при встрече обменялись рукопожатиями (каждый пожал руку каждому по одному разу). Сколько всего рукопожатий было сделано?

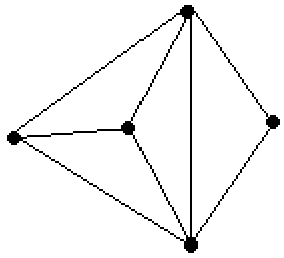
38. К XVIII веку через реку, на которой стоял город Кенигсберг (ныне Калининград), было построено 7 мостов, которые связывали с берегами и друг с другом два острова, расположенные в пределах города.
Задача заключается в следующем: нужно пройти (если это возможно) по всем семи мостам так, чтобы на каждом из них побывать лишь по одному разу и вернуться к тому месту, откуда начал маршрут.

39. В трех различных домах живут три поссорившиеся между собой соседа. Недалеко от их домов имеются три колодца. Можно ли от каждого дома проложить к каждому из колодцев тропинку так, чтобы никакие две из них не пересекались?

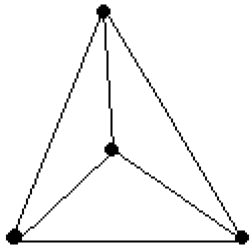
40. В городе Н от каждой площади отходит ровно пять улиц, соединяющих площади. Докажите, что число площадей чётно, а число улиц кратно пяти.

41. В государстве система авиалиний устроена таких образом, что любой город соединён авиалиниями не более чем с тремя другими и из любого города в любой другой можно проехать, сделав не более одной пересадки. Какое максимальное число городов может быть в этом государстве?

42. Можно ли нарисовать графы, изображенные на рисунках, не отрывая карандаш от бумаги и проводя каждое ребро ровно один раз?

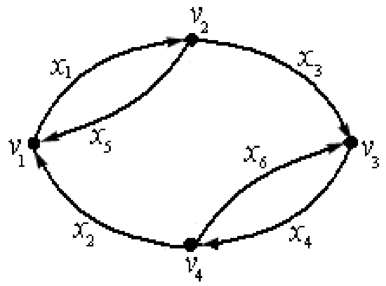


1)

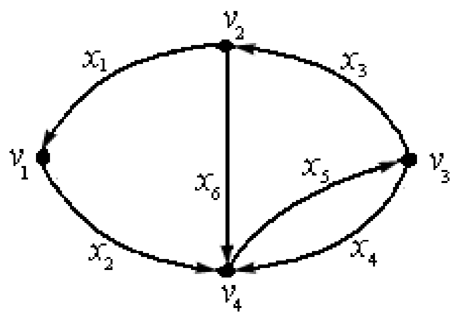


2)

43. Составить матрицу инцидентности данного орграфа:



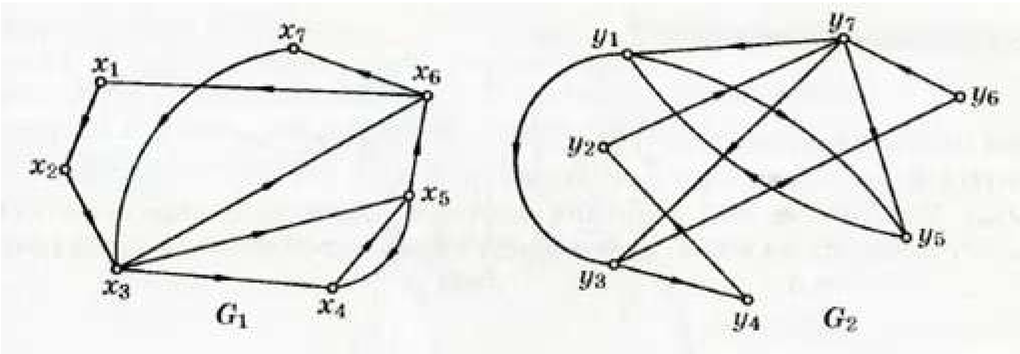
44. Составить матрицу смежности данного орграфа:



45. Может ли так случиться, что в одной компании из шести человек каждый знаком с двумя и только с двумя другими?

46. Из пункта А в пункт В выехали пять машин одной марки разного цвета: белая, черная, красная, синяя, зеленая. Черная едет впереди синей, зеленая – впереди белой, но позади синей, красная впереди черной. Какая машина едет первой и какая последней?

47. Пусть даны графы $G_1(X, E)$ и $G_2(Y, E)$. Установите, изоморфны ли данные графы



48. Найдите функции **g** и **h** в рекурсивной формуле для двухместной функции **f(x,y)=x y** , если рекурсия проводиться по переменной **x.**

49. Найдите функции **g** и **h** в рекурсивной формуле для трехместной функции **f(x,y,z) = x y+z,** если рекурсия проводится по переменной **y.**

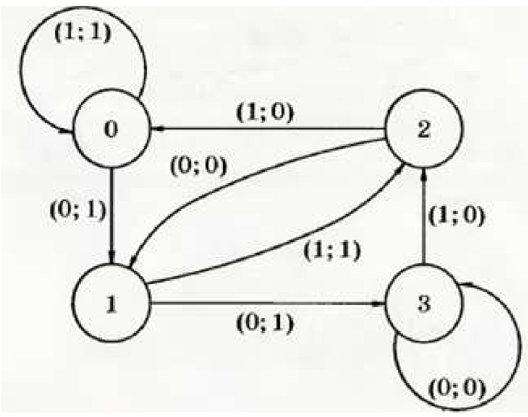
50. Примените оператор минимизации к функции

$$f(x) = \begin{cases} 3x + 2, \text{если x} \neq 2 \\ \text{не определено, если x} = 2 \end{cases}$$

51. Для автомата, заданного таблицей, постройте диаграмму Мура. Задайте этот автомат системой булевых функций

$\alpha \backslash q$	0	1	2	3
0	(1;1)	(3;0)	(2;0)	(2;0)
1	(2;1)	(2;0)	(3;0)	(3;0)

52. Для автомата, заданного диаграммой Мура, выпишите соответственную таблицу и систему булевых функций



53. Для автомата, заданного каноническими уравнениями, постройте диаграмму Мура и соответствующую таблицу $\begin{cases} z(t+1) = x(t), \\ y(t) = x(t) \leftrightarrow z(t) \end{cases}$.

5. Описание процедуры проведения промежуточной аттестации

Зачет проводится за счет часов, отведенных на дисциплину, в учебное время по вопроснику, согласованному на ЦК и утвержденному заведующим отделением СПО.

Обучающийся получает одно теоретическое и одно практическое задания. На подготовку к ответу обучающемуся отводится до 20 минут. Обучающийся предъявляет ответы в смешанной форме: устно раскрывает теоретические вопросы; решение задач представляется в письменном виде с устными комментариями (пояснениями).

6. Эталоны ответов.

6.1. Эталоны ответов на устные вопросы:

1. Математика стала частью нашей культуры. Человек не может считать себя широкообразованным, не имея представления современной математике, ее роли в повседневной жизни, в науке. Основу составляют такие разделы математики, как комбинаторный анализ, теории множеств, элементы математической и классической логики. Дискретная математика для программистов относится к числу общепроффесиональных предметов, формирующих базовый уровень знаний, необходимых для изучения других дисциплин, таких, как «Математическая статистика», «архитектура ЭВМ, систем сетей», «Базы данных», «Компьютерное моделирование», «Автоматизированные системы», «Технология разработки программных продуктов», «Основы алгоритмизации и программирования». Предложенный вариант учебной дисциплины можно назвать «Курсом на межпредметной основе». Для этого математические знания были увязаны с общенаучными и учтены современные представления о функциональной роли математического образования. Курс формирует у учащихся систему умений и навыков самостоятельного избирательного восприятия информации и ее переработки. Его задача – научить систематизации, обобщению, структурированию знаний, а так же их адекватному применению как в предметных областях, так и в практической деятельности. Сочетание фундаментальных теоретических знаний с их функциональной направленностью призвано показать учащимся использование универсального математического аппарата применительно к различным предметным областям и разнообразным видам деятельности. Акцент делается на знакомство с разными приемами систематизации знаний и предоставлении информации в сжатом виде.

2. Составные высказывания. Простейшие связки. Логические отношения.

Высказыванием называется повествовательное предложение, о котором в данной ситуации можно сказать, что оно истинно или ложно, но не то и другое одновременно.

Приведем примеры высказываний.

Пример 1. Волга впадает в Каспийское море.

Пример 2. Два больше трех.

Первое высказывание является истинным, а второе — ложным.

Таким образом, высказывание обладает свойством представлять истину или ложь, поэтому на высказывание можно смотреть как на величину, которая может принимать только одно из двух значений: «истина», «ложь».

Поставим в соответствие высказыванию логическую переменную x, которая принимает значение 1, если высказывание истинно, и 0, если высказывание ложно.

Это новое высказывание называется *составным*, в то время как высказывания, из которых оно образовано, называются его простыми составляющими или компонентами. Любое высказывание, даже такое, которое на самом деле является сложным, может быть использовано в качестве одного из простых составляющих какого-то другого составного высказывания.

Составные высказывания будем получать из простых с помощью логических операций: *отрицание, конъюнкция, дизъюнкция, импликация, эквивалентность*, которые осуществляются при помощи логических связок: $\overline{}$; $\wedge$; $\vee$; $\rightarrow$; $\leftrightarrow$.

Название	Прочтение	Обозначение
Отрицание	не	$-$
Конъюнкция	и	$\wedge$
Дизъюнкция	или	$\vee$
Импликация	если...то	$\rightarrow$
Эквивалентность	тогда и только тогда, когда	$\leftrightarrow$

При рассмотрении той или иной связки мы хотим знать, каким именно образом истинность составного высказывания, порожденного этой связкой, зависит от истинности его компонент. Очень удобно изображать эту зависимость, пользуясь таблицами истинности, которые называются также интерпретациями логических операций. Каждой строке таблицы истинности взаимно однозначно соответствует набор составляющих высказываний и соответствующее значение составного высказывания. Наборы из нулей и единиц, соответствующих составляющим высказываниям, в каждой строке таблицы истинности имеют стандартное расположение, т. е. расположены в лексикографическом порядке (порядке возрастания).

Отрицанием высказывания X называется высказывание $\overline{X}$, которое истинно, когда X ложно, и ложно, когда X истинно.

Таблица истинности для отрицания.

X	$\overline{X}$
0	1
1	0

Конъюнкцией двух высказываний X и Y называется высказывание $X \wedge Y$, которое истинно только в том случае, когда X и Y оба истинны.

Таблица истинности для конъюнкций.

X	Y	$X \wedge Y$
0	0	0
0	1	0
1	0	0
1	1	1

Дизъюнкцией двух высказываний X и Y называется высказывание $X \vee Y$, которое истинно, когда хотя бы одно из них истинно.

Таблица истинности дизъюнкций.

X	Y	$X \vee Y$
0	0	0
0	1	1
1	0	1
1	1	1

Импликацией двух высказываний X и Y называется высказывание $X \to Y$, которое ложно тогда и только тогда, когда X истинно, а Y ложно.

Таблица истинности для импликации.

X	Y	$X \to Y$
0	0	1
0	1	1
1	0	0
1	1	1

Эквивалентностью высказываний X и Y называется высказывание $X \leftrightarrow Y$, которое истинно тогда и только тогда, когда X и Y оба истинны или ложны.

Таблица истинности для эквивалентности.

X	Y	$X \leftrightarrow Y$
0	0	1
0	1	0
1	0	0
1	1	1

Новые высказывания могут быть образованы при помощи нескольких логических операций и составлять формулы, некоторые из которых рассматриваются как логические операции, осуществляемые при помощи других логических связок: |; ↓; ⊕.

Название	Прочтение	Обозначение
Штрих Шеффера	Антиконъюнкция	|
Стрелка Пирса	Антидизъюнкция	↓
Сумма по модулю два	Антиэквивалентность	⊕

Штрих Шеффера, $X \mid Y$ или антиконъюнкция, по определению $(X \mid Y) = \overline{X \wedge Y}$.

Таблица истинности штриха Шеффера.

X	Y	$X \mid Y$
0	0	1
0	1	1
1	0	1
1	1	0

Стрелка Пирса, или антидизъюнкция, по определению $X \downarrow Y = \overline{X \vee Y}$.

Таблица истинности стрелки Пирса.

X	Y	$X \downarrow Y$
0	0	1
0	1	0
1	0	0
1	1	0

Сумма по модулю два, или антиэквивалентность, по определению $X \oplus Y = \overline{X \leftrightarrow Y}$.

Таблица истинности суммы по модулю два.

X	Y	$X \oplus Y$
0	0	0
0	1	1
1	0	1
1	1	0

Заметим, что таблицы истинности логических операций содержат 2^n строк, где n — число простых высказываний.

3. Варианты импликации.

Импликация двух высказываний отличается от эквивалентности, а также от дизъюнкции и конъюнкции тем, что она несимметрична. Так $X \vee Y$ эквивалентно $Y \vee X$; $X \wedge Y$ эквивалентно $Y \wedge X$; $X \leftrightarrow Y$ эквивалентно $Y \leftrightarrow X$, но $X \to Y$ не эквивалентно $Y \to X$. Высказывание $Y \to X$ называется **конверсией** высказывания $X \to Y$. Многие из наиболее распространенных ошибок в рассуждениях происходят от смешивания какого-либо высказывания с его конверсией. Интересно поэтому рассмотреть те импликации, которые могут быть образованы из высказываний X и Y.

В таблице истинности представлены четыре импликации и их названия.

X	Y	$\overline{X}$	$\overline{Y}$	Импликация $X \to Y$	Конверсия импликации $Y \to X$	Конверсия контрапозиции $\overline{X} \to \overline{Y}$	Контрапозиция $\overline{Y} \to \overline{X}$
0	0	1	1	1	1	1	1
0	1	1	0	1	0	0	1
1	0	0	1	0	1	1	0
1	1	0	0	1	1	1	1

Из таблицы видно, что $X \to Y$ эквивалентно $\overline{Y} \to \overline{X}$. Последнее называется контрапозицией первого. Контрапозиция является удобной формой импликации во многих рассуждениях. Аналогично, высказывание $\overline{X} \to \overline{Y}$ представляет собой конверсию контрапозиции. Так как контрапозиция эквивалентна $\overline{X} \to \overline{Y}$, то конверсия этой контрапозиции эквивалентна конверсии этой импликации.

С импликацией связано постоянное упоминание математиками «необходимое условие» и «достаточное условие».

X является достаточным условием для Y	Если имеет место X, то Y также будет иметь место	Импликация $X \to Y$
X является необходимым условием для Y	Если имеет место Y, то X также будет иметь место	Конверсия достаточного условия $Y \to X$
X является необходимым и достаточным условием для Y	X имеет место, если и только если имеет место Y	Двойная импликация $X \leftrightarrow Y$ эквивалентность

4. Основные законы, определяющие свойства логических операций.

1) Идемпотентность дизъюнкции и конъюнкции:

$$X \vee X \leftrightarrow X, \quad X \wedge X \leftrightarrow X.$$

2) Коммутативность дизъюнкции и конъюнкции:

$$X \vee Y \leftrightarrow Y \vee X, \quad X \wedge Y \leftrightarrow Y \wedge X.$$

3) Ассоциативность дизъюнкции и конъюнкции:

$$X \vee (Y \vee Z) \leftrightarrow (X \vee Y) \vee Z, \quad X \wedge (Y \wedge Z) \leftrightarrow (X \wedge Y) \wedge Z.$$

4) Дистрибутивность операций дизъюнкции и конъюнкции относительно друг друга:

$$X \vee (Y \wedge Z) \leftrightarrow (X \vee Y) \wedge (X \vee Z), \quad X \wedge (Y \vee Z) \leftrightarrow (X \wedge Y) \vee (X \wedge Z).$$

5) Двойное отрицание:

$$X \leftrightarrow \overline{\overline{X}}.$$

6) Закон де Моргана:

$$\overline{X} \vee \overline{Y} \leftrightarrow \overline{X \wedge Y}, \quad \overline{X} \wedge \overline{Y} \leftrightarrow \overline{X \vee Y}.$$

7) Склеивание:

$$(X \vee Y) \wedge (X \vee \overline{Y}) \leftrightarrow X, \quad (X \wedge Y) \vee (X \wedge \overline{Y}) \leftrightarrow X.$$

8) Поглощение:

$$X \vee (X \wedge Y) \leftrightarrow X, \quad X \wedge (X \vee Y) \leftrightarrow X.$$

9) Действие с логическими константами 0 и 1:

$$X \vee 0 \leftrightarrow X, \quad X \wedge 0 \leftrightarrow 0, \quad X \vee 1 \leftrightarrow 1, \quad X \wedge 1 \leftrightarrow X, \quad X \wedge \overline{X} \leftrightarrow 0.$$

10) Закон исключения третьего:

$$X \vee \overline{X} \leftrightarrow 1.$$

11) Тождество:

$$X \leftrightarrow X.$$

12) Отрицание противоречия:

$$\overline{\overline{X} \wedge X} \leftrightarrow 1.$$

13) Контрапозиция:

$$(X \to Y) \leftrightarrow (\overline{Y} \to \overline{X}).$$

14) Цепное заключение:

$$((X \to Y) \wedge (Y \to Z)) \leftrightarrow (X \to Z).$$

15) Противоположность:

$$(X \leftrightarrow Y) \leftrightarrow (\overline{X} \leftrightarrow \overline{Y}).$$

16) Модус поненс (modus ponens):

$$(X \wedge (X \to Y)) \to Y \leftrightarrow 1.$$

Сформулированные законы легко проверить с помощью таблицы истинности.

Заметим, что при исследовании различных высказываний на эквивалентность (равносильность) логическую связку $\leftrightarrow$ можно заменить обычным знаком равенства =.

5. Булевы функции.

Булева функция, или функция алгебры логики, является одним из основных объектов дискретной математики.

Функцию $f(x_1, x_2, \ldots, x_n)$, принимающую одно из двух значений 0 или 1, от n переменных, каждая из которых принимает одно из двух значений 0 или 1, будем называть булевой функцией $f(x_1, x_2, \ldots, x_n)$ от n переменных.

Булева функция от n переменных сопоставляет каждому упорядоченному набору (кортежу), составленному из n элементов, 0 и 1, либо 1, либо 0.

Две булевы функции называются равными, если для любых одинаковых наборов значений переменных обе функции принимают одинаковые значения. Булевых функций одной переменной четыре, а двух переменных — шестнадцать и т. д. Число булевых функций от n переменных равно 2^{2^n}.

Таблица истинности булевой функции одной переменной:

x	$f_1(x)$	$f_2(x)$	$f_3(x)$	$f_4(x)$
0	0	0	1	1
1	0	1	0	1

Функции $f_1(x)$ и $f_4(x)$) называются константами — соответственно 0 и 1.

Функция $f_2(x)$ совпадает с переменной x и называется тождественной $f_2(x) = x$:

Функция $f_3(x)$ принимает значения, противоположные значениям аргумента x, и называется отрицанием x, обозначается $\overline{x}$: $f_3(x) = \overline{x}$.

Таблица истинности булевой функции двух переменных:

x_1	x_2	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	f_{10}	f_{11}	f_{12}	f_{13}	f_{14}	f_{15}	f_{16}
0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
0	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
1	0	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

Следует отметить, что здесь к функциям двух переменных относятся и такие, которые в действительности зависят от одной переменной.

1. Функции f_1 и f_{16} представляют собой константы 0 и 1.

2. Функции f_4, f_6, f_{11}, f_{13} существенно зависят только от одной переменной: $f_4 = x_1$, $f_6 = x_2$, $f_{11} = \overline{x_2}$, $f_{13} = \overline{x_1}$.

3. Остальные функции существенно зависят от двух переменных, и для них есть названия и обозначения:

а) функция $f_2 = x_1 \wedge x_2$ и называется конъюнкцией,

б) функция $f_8 = x_1 \vee x_2$ и называется дизъюнкцией,

в) функция $f_{10} = x_1 \leftrightarrow x_2$ и называется эквивалентностью,

г) функция $f_7 = x_1 \oplus x_2$ и называется суммой по модулю два, или суммой Жегалкина,

д) функция $f_{12} = x_2 \to x_1$ и называется конверсией,

е) функция $f_{14} = x_1 \to x_2$ и называется импликацией,

ж) функция $f_{15} = x_1 \mid x_2$ и называется штрих Шеффера,

з) функция $f_9 = x_1 \downarrow x_2$ и называется стрелкой Пирса,

и) функции f_3 и f_5 логически несовместимы с импликацией и конверсией и называются функциями запрета.

6. Свойства элементарных булевых функций.

Для булевых функций справедливы равенства, аналогичные формулам, сформулированным для высказываний.

1. Функции: конъюнкция, дизъюнкция, сумма по модулю два, стрелка Пирса, штрих Шеффера обладают свойством коммутативности.

2. Функции: конъюнкция, дизъюнкция, сумма по модулю два обладают свойством ассоциативности и свойством дистрибутивности.

3. Закон де Моргана: $\overline{x_1 \wedge x_2} = \overline{x_1} \vee \overline{x_2}$; $\overline{x_1 \vee x_2} = \overline{x_1} \wedge \overline{x_2}$.

4. Закон двойного отрицания: $\overline{\overline{x}} = x$.

5. Выражение дизъюнкции через конъюнкцию и суммы по модулю два: $x_1 \vee x_2 = x_1 \wedge x_2 \oplus x_2 \oplus x_1$.

6. Выражение дизъюнкции через импликацию:

$$(x_1 \to x_2) \to x_2 = x_1 \vee x_2.$$

7. Выражение отрицания через штрих Шеффера, стрелку Пирса, сумму по модулю два и эквивалентность: $x \mid x = x \downarrow x = \overline{x} = x \oplus 1 = x \leftrightarrow 0$.

8. Выражение конъюнкции через штрих Шеффера:

$$(x_1 \mid x_2) \mid (x_1 \mid x_2) = x_1 \wedge x_2.$$

9. Выражение дизъюнкции через стрелку Пирса:

$$(x_1 \downarrow x_2) \downarrow (x_1 \downarrow x_2) = x_1 \vee x_2.$$

10. Закон поглощения: $x_1 \wedge x_2 \vee x_1 = x_1$; $x_1 \wedge (x_1 \vee x_2) = x_1$.

11. Закон склеивания: $\overline{x} \vee x = \overline{x} \oplus x = 1$.

12. Для функций: конъюнкция, дизъюнкция и сумма по модулю два справедливы следующие тождества:

$$x \wedge x = x; \quad x \vee x = x; \quad x \oplus x = 0;$$
$$\overline{x} \wedge x = 0; \quad \overline{x} \vee x = 1; \quad x \oplus \overline{x} = 1;$$
$$x \wedge 0 = 0; \quad x \vee 0 = x; \quad x \oplus 0 = x;$$
$$x \wedge 1 = x; \quad x \vee 1 = 1; \quad x \oplus 1 = \overline{x}.$$

13. Для функций конъюнкция и дизъюнкция справедливы тождества: $x_1 \vee \overline{x_1} \wedge x_2 = x_1 \vee x_2$; $x_1 \wedge (\overline{x_2} \vee x_1) = x_1 \wedge x_2$.

Для доказательства справедливости любых из приведенных тождеств нужно составить таблицы истинности для булевых функций.

Булеву функцию любого числа переменных можно задать формулой, содержащей функции одной и двух переменных посредством подстановки одних булевых функций вместо переменных в другие булевы функции, т. е. посредством суперпозиции булевых функций.

7. Дизъюнктивные и конъюнктивные нормальные формы алгебры высказываний.

Конъюнктивным одночленом от переменных $x_1, x_2, \ldots, x_n$ называется конъюнкция этих переменных или их отрицаний.

Дизъюнктивным одночленом от переменных $x_1, x_2, \ldots, x_n$ называется дизъюнкция этих переменных или их отрицаний.

Формула, равносильная данной формуле алгебры высказываний и являющаяся дизъюнкцией элементарных конъюнктивных одночленов, называется **дизъюнктивной нормальной формой** (ДНФ) данной формулы.

Например: $(x_1 \wedge x_2 \wedge x_3) \vee (x_1 \wedge \overline{x_2}) \vee (x_3 \wedge x_2) \vee \overline{x_3}$ — ДНФ.

Формула, равносильная данной формуле алгебры высказываний и являющаяся конъюнкцией элементарных дизъюнктивных одночленов, называется **конъюнктивной нормальной формой** (КНФ) данной формулы.

Например: $(\overline{x_1} \vee x_2 \vee x_3) \wedge (x_1 \vee \overline{x_3}) \wedge x_2$ — КНФ.

Для каждой формулы алгебры высказываний можно найти множество дизъюнктивных и конъюнктивных нормальных форм.

Алгоритм построения

(1) Избавиться от всех логических операций, содержащихся в формуле, заменив их основными: конъюнкцией, дизъюнкцией, отрицанием. Это можно сделать, используя равносильные формулы:

$$x_1 \to x_2 = \overline{x_1} \vee x_2; \quad x_1 \leftrightarrow x_2 = (\overline{x_1} \vee x_2) \wedge (x_1 \vee \overline{x_2});$$
$$x_1 \leftrightarrow x_2 = (x_1 \wedge x_2) \vee (\overline{x_1} \wedge \overline{x_2}).$$

(2) Заменить знак отрицания, относящийся к выражениям типа $\overline{x_1 \wedge x_2}$ или $\overline{x_1 \vee x_2}$, знаками отрицания, относящимися к отдельным переменным высказываниям на основании формул:

$$\overline{x_1 \vee x_2} = \overline{x_1} \wedge \overline{x_2}; \quad \overline{x_1 \wedge x_2} = \overline{x_1} \vee \overline{x_2}.$$

(3) Избавиться от знаков двойного отрицания.

(4) Применить, если нужно, к операциям конъюнкции и дизъюнкции свойства дистрибутивности и формулы поглощения.

8. Совершенная дизъюнктивная и совершенная конъюнктивная нормальные формы.

Любая булева функция может иметь много представлений в виде ДНФ и КНФ. Особое место среди этих представлений занимают совершенные ДНФ (СДНФ) и совершенные КНФ (СКНФ).

Совершенная дизъюнктивная нормальная форма (СДНФ) — это ДНФ, в которой в каждый конъюнктивный одночлен каждая переменная x_i из набора $f(x_1, x_2, \ldots, x_n)$ входит ровно один раз, причем входит либо сама x_i, либо ее отрицание $\overline{x_i}$.

Конструктивно СДНФ для каждой формулы алгебры высказываний, приведенной к ДНФ, можно определить так:

Совершенной дизъюнктивной нормальной формой (СДНФ) формулы алгебры высказываний называется ее ДНФ, обладающая следующими свойствами:

1. ДНФ не содержит двух одинаковых конъюнкций.
2. Ни одна конъюнкция не содержит одновременно двух одинаковых переменных.
3. Ни одна конъюнкция не содержит одновременно некоторую переменную и ее отрицание.
4. Каждая конъюнкция содержит либо переменную x_i, либо ее отрицание $\overline{x_i}$ для всех переменных, входящих в формулу.

Конструктивно СКНФ для каждой формулы алгебры высказываний, приведенной к КНФ, можно определить так:

Совершенной конъюнктивной нормальной формой (СКНФ) данной формулы алгебры высказываний называется такая ее КНФ, которая удовлетворяет следующим свойствам:

1. КНФ не содержит двух одинаковых дизъюнкций.
2. Ни одна из дизъюнкций не содержит одновременно двух одинаковых переменных.
3. Ни одна из дизъюнкций не содержит одновременно некоторую переменную и ее отрицание.
4. Каждая дизъюнкция СКНФ содержит либо переменную x_i, либо ее отрицание x_i для всех переменных, входящих в формулу.

Теорема 1. Произвольную булеву функцию $f(x_1, x_2, \ldots, x_n)$ можно задать формулой $f(x_1, x_2, \ldots, x_n) = \bigvee(x_1^{\tau_1} \wedge x_2^{\tau_2} \wedge \ldots \wedge x_n^{\tau_n})$, где дизъюнкция берется по всем $\tau = (\tau_1, \tau_2, \ldots, \tau_n)$, где $f(\tau) = 1$ и

$$x_i = \begin{cases} \overline{x_i}, & \text{если } \tau = 0, \\ x_i, & \text{если } \tau = 1. \end{cases}$$

Теорема 2. Произвольную булеву функцию $f(x_1, x_2, \dots, x_n)$ можно задать формулой $f(x_1, x_2, \dots, x_n) = \bigwedge(x_1^{\overline{\tau_1}} \vee x_2^{\overline{\tau_2}} \vee \dots \vee x_n^{\overline{\tau_n}})$, где конъюнкция берется по всем $\overline{\tau} = (\overline{\tau_1}, \overline{\tau_2}, \dots, \overline{\tau_n})$, где $f(\overline{\tau}) = 0$ и

$$x_i = \begin{cases} x_i, & \text{если } \overline{\tau} = 0, \\ \overline{x_i}, & \text{если } \overline{\tau} = 1. \end{cases}$$

Эти формулы называются соответственно **совершенной дизъюнктивной нормальной формой** или **совершенной конъюнктивной нормальной формой** булевой функции $f(x_1, x_2, \dots, x_n)$. Исходя из таблицы истинности булевой функции, можно построить СДНФ функции: для каждого набора $\tau = (\tau_1, \tau_2, \dots, \tau_n)$, такого, что $f(\tau) = 1$, составляется конъюнкция $x_1^{\tau_1} \wedge x_2^{\tau_2} \wedge \dots \wedge x_n^{\tau_n}$, а затем все эти конъюнкции соединяем знаком дизъюнкции.

Для построения СКНФ функции выписываем наборы $\overline{\tau} = (\overline{\tau_1}, \overline{\tau_2}, \dots, \overline{\tau_n})$ такие, что $f(\overline{\tau}) = 0$. Для такого набора составляется дизъюнкция

$$x_1^{\overline{\tau_1}} \vee x_2^{\overline{\tau_2}} \vee \dots \vee x_n^{\overline{\tau_n}},$$

а затем все такие дизъюнкции соединяют знаком конъюнкции.

Приведенные формулы позволяют сформулировать следующие утверждения:

1. Каждая булева функция от n переменных, отличная от константы 0, имеет единственную СДНФ.

2. Каждая булева функция от n переменных, отличная от константы 1, имеет единственную СКНФ.

Эти утверждения называются **теоремой о функциональной полноте**.

9. Многочлены Жегалкина.

Многочленом Жегалкина называется многочлен, являющийся суммой константы и различных одночленов, в которые каждая из переменных входит не выше, чем в первой степени.

Многочлен Жегалкина константы равен самой же константе; многочлен Жегалкина булевой функции одной переменной $f(x) = a_0 \oplus a_1 x$; многочлен Жегалкина булевой функции двух переменных

$$f(x_1, x_2) = a_0 \oplus a_1 x_1 \oplus a_2 x_2 \oplus a_{12}(x_1 \wedge x_2);$$

многочлен Жегалкина булевой функции трех переменных

$$f(x_1, x_2, x_3) = a_0 \oplus a_1 x_1 \oplus a_2 x_2 \oplus a_3 x_3 \oplus \\ \oplus a_{12}(x_1 \wedge x_2) \oplus a_{13}(x_1 \wedge x_3) \oplus a_{23}(x_2 \wedge x_3) \oplus a_{123}(x_1 \wedge x_2 \wedge x_3)$$

и т. д. Коэффициенты $a_{1,\ldots,i}$ и свободный член a_0 принимают значения 0 или 1, а число слагаемых в формуле равно 2^n, где n — число переменных. Знак $\oplus$ — сумма Жегалкина или сумма по модулю два.

Теорема 3 (Жегалкина). Каждая булева функция $f(x_1, x_2, \ldots, x_n)$ может быть представлена в виде многочлена Жегалкина и притом единственным образом, с точностью до порядка слагаемых.

1. Находим множество тех двоичных наборов, на которых функция принимает значение 1.

2. Составляем СДНФ.

3. В СДНФ каждый знак дизъюнкции меняем на знак суммы Жегалкина.

4. Упрощаем, если можно, полученное выражение, используя тождество $\overline{x_i} \oplus x_i = 1$.

5. В полученной формуле каждое отрицание $\overline{x_i}$ заменяем на $x_i \oplus 1$.

6. Раскрываем скобки в полученной формуле, содержащей только функции $\wedge$ и $\oplus$ и константу 1.

7. Приводим подобные члены, используя тождество $x_i \oplus x_i = 0$.

Используя метод неопределенных коэффициентов, получаем второй алгоритм определения многочлена Жегалкина:

1. Составляем систему линейных уравнений относительно 2^n неизвестных коэффициентов, содержащую 2^n уравнений, решением которой являются коэффициенты $a_0, a_1, \ldots, a_{1,2,\ldots,n}$ многочлена Жегалкина.

Многочлен Жегалкина называется **нелинейным**, если он содержит конъюнкции переменных, а если он не содержит конъюнкции переменных, то он называется **линейным**.

Функция $f(x_1, x_2, \ldots, x_n)$ называется линейной, если ее многочлен Жегалкина имеет вид $a_0 \oplus a_1 x_1 \oplus \ldots \oplus a_n x_n$, и нелинейной в противном случае.

Из определения многочлена Жегалкина следует, что для любой булевой функции коэффициенты при переменных $x_1, x_2, \ldots, x_n$ и свободный член вычисляются по формулам:

$$
\begin{gathered}
a_0 = f(0, \ldots, 0),\\
a_1 = f(0, \ldots, 0) \oplus f(1, \ldots, 0),\\
a_2 = f(0, \ldots, 0) \oplus f(0, 1, \ldots, 0),\\
\ldots\\
a_n = f(0, \ldots, 0) \oplus f(0, \ldots, 1).
\end{gathered}
$$

На этом основан алгоритм определения линейности (или нелинейности) булевой функции.

1. По таблицам истинности булевой функции $f(x_1, x_2, \ldots, x_n)$ и выше указанным формулам находим коэффициенты: $(a_0, a_1, \ldots, a_n)$.

2. Выписываем многочлен $\Phi(x_1, \ldots, x_n) = a_0 \oplus a_1 x_1 \oplus \ldots \oplus a_n x_n$ и проверяем, задает ли он эту функцию. Для этого строим таблицу истинности многочлена $\Phi(x_1, x_2, \ldots, x_n)$ и сравниваем ее с таблицей истинности функции $f(x_1, x_2, \ldots, x_n)$.

Если таблицы истинности совпадают, то функция $f(x_1, x_2, \ldots, x_n)$ линейная и $\Phi(x_1, x_2, \ldots, x_n)$ — ее многочлен Жегалкина. В противном случае функция нелинейная.

10. Специальные классы булевых функций: функции, сохраняющие единицу, функции, сохраняющие нуль, самодвойственные функции, линейные функции, монотонные функции. Теорема Поста о функциональной полноте.

Определение. Множество M логических функций называется *замкнутым классом*, если любая суперпозиция функций из множества M снова принадлежит M.

Всякая система Σ логических функций порождает некоторый замкнутый класс, а именно класс всех функций, которые можно получить суперпозициями функций Σ. Такой класс называется *замыканием* Σ и обозначается $[\Sigma]$. Если множество M - функционально полная система, то $[M] = P_2$.

Теорема. Всякая булева формула, не содержащая отрицаний, представляет собой монотонную функцию отличную от константы; наоборот, для любой монотонной функции, отличной от 0 и 1, найдётся представляющая её булева формула без отрицаний.

Из данной теоремы и того очевидного факта, что подстановка нескольких формул без отрицаний в формулу без отрицаний снова даёт формулу без отрицаний, вытекает следующая теорема.

Теорема. Множество всех монотонных функций является замкнутым классом.

Но поскольку всякая булева формула без отрицаний является суперпозицией дизъюнкций и конъюнкций, из данной теоремы непосредственно получаем следствие.

Следствие. Класс монотонных функций является замыканием системы функций

Проверка булевой функции на принадлежность к замкнутым классам, на полноту.

Теорема. Если все функции функционально полной системы Σ^* представимы формулами над системой Σ, то система Σ также функционально полна.

Пример. а) Системы $\Sigma_1 = \{\wedge, \bar{\ }\}$ и $\Sigma_2 = \{\vee, \bar{\ }\}$ функционально полны. Действительно, с помощью законов Де Моргана и двойного отрицания можно выразить в каждой из этих систем функцию, недостающую до Σ_0 через остальные две:

$$x_1 \vee x_2 = \overline{\bar{x}_1 \wedge \bar{x}_2},\ x_1 \wedge x_2 = \overline{\bar{x}_1 \vee \bar{x}_2}.$$

С точки зрения функциональной полноты систему Σ_0 следует считать избыточной: она сохраняет свойство полноты и при удалении из неё конъюнкции или дизъюнкции. Однако легко видеть из приведённого примера, что, хотя системы Σ_1 и Σ_2 не являются избыточными, зато формулы в них получаются гораздо длиннее: замена одной операции на другую вносит в формулу сразу три лишних отрицания.

б) Системы $\Sigma_3 = \{\,|\,\}$ (штрих Шеффера) и $\Sigma_4 = \{\downarrow\}$ (стрелка Пирса) являются функционально полными.

$$\bar{x} = x | x = x \downarrow x;\ x_1 \vee x_2 = (x_1 \downarrow x_2) \downarrow (x_1 \downarrow x_2);\ x_1 \wedge x_2 = (x_1 | x_2) | (x_1 | x_2).$$

Таким образом, система Σ_3 сводится к системе Σ_1, а система Σ_4 - к системе Σ_2.

в) Система $\Sigma_5 = \{\otimes, \oplus, 1\}$ ($\otimes$ – умножение по модулю 2, $\oplus$ – сложение по модулю 2 – см. пункт 1 лекции № 8)) является функционально полной. Поскольку $\bar{x} = x \oplus 1$, данная система сводится к Σ_0.

11. Понятие множества. Способы задания множества. Подмножества. Операции над множествами.

Имеется два существенно различных способа задания множеств. Можно либо указать правило для определения того, принадлежит или не принадлежит рассматриваемому множеству любой данный объект, либо дать полный перечень элементов этого множества.

Первый способ мы назовем описанием множества, а второй способ — перечислением множества. Например, обозначение $\{x \in U : \alpha(x)\}$ читается: «элементы множества U, обладающие свойством α» — это описание множества. Элементы перечисляемого множества принято заключать в скобки: $\{1, 2, 3, \dots\}$ — множество натуральных чисел; $\{2, 4, 6, \dots\}$ — множество четных чисел. Под многоточием в первом случае подразумеваются все последующие натуральные числа, а во втором — четные.

Множество, состоящее из некоторых элементов другого множества, называется подмножеством этого последнего множества. С целью изучения всех подмножеств данного множества введем следующую терминологию. Исходное множество будем называть *универсальным множеством*; подмножества, содержащие один элемент, будем называть *единичными множествами*; множество, вовсе не содержащее никаких элементов, будем называть *пустым множеством* и обозначать $\varnothing$.

В качестве примера возьмем универсальное множество U, состоящее из трех элементов $\{a, b, c\}$. Собственные подмножества U — это множества, которые содержат некоторые, но не все элементы U. Этими подмножествами являются три множества из двух элементов $\{a, b\}$, $\{a, c\}$, $\{b, c\}$ и три единичных множества $\{a\}$, $\{b\}$, $\{c\}$.

Будем считать подмножеством множества U и пустое множество $\varnothing$, не содержащее элементов U.

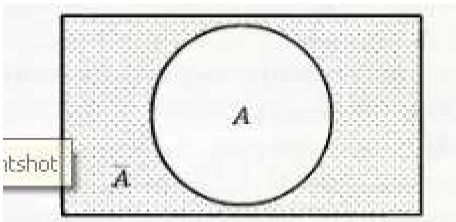
Другими словами, множество A называется подмножеством множества B (обозначаем $A \subset B$), если все элементы множества A принадлежат B. Это означает справедливость следующего утверждения: для любого элемента a, если $a \in A$, то $a \in B$ при условии $A \subset B$. Будем говорить также, что множество A содержится в B или имеется включение множества A в B. Множества A и B называются равными или совпадающими (обозначается $A = B$), если они состоят из одних и тех же элементов, т. е. если $A \subset B$ и $B \subset A$. Таким образом, чтобы доказать равенство множеств, требуется установить два включения.

Следует провести аналогию между логическими операциями и операциями над множествами.

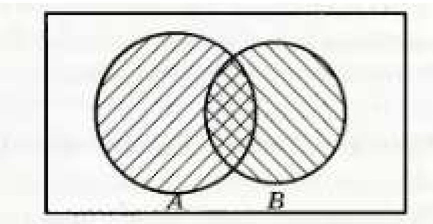
отрицание	дополнение
конъюнкция	пересечение
дизъюнкция	объединение
импликация	разность

Чтобы нагляднее представить эти операции, изобразим их на диаграмме, называемой диаграммой Эйлера–Венна. Пусть прямоугольник обозначает универсальное множество, а круги внутри прямоугольника — подмножества.

Дополнением к множеству A называется множество элементов, которые не содержатся в A. Обозначают его $\overline{A}$ и читают «дополнение множества A к U»



Пересечением множеств A и B называется множество элементов,
надлежащих и A и B. Обозначают $A \cap B$ и читают «пересечение A и B»



Если A и B — непустые множества, пересечение которых пусто, т. е. $A \cap B = \varnothing$, то их называют непересекающимися множествами.

Объединением множеств A и B называется множество элементов, принадлежащих либо A, либо B (либо обоим). Обозначают $A \cup B$ и читают «объединение A и B» (см. рис. 2.3).

Разностью множеств A и B называется множество элементов, принадлежащих A и не принадлежащих B. Обозначают $A \setminus B$ и читают «разность A и B» (см. рис. 2.4).

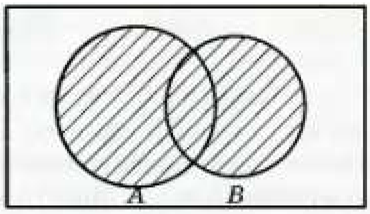


Рис. 2.3

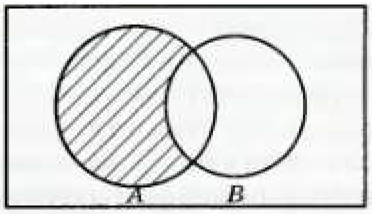


Рис. 2.4

12. Соотношения между множествами и составными высказываниями.

Существует тесная связь между множествами — с одной стороны, и высказываниями — с другой, а также между операциями над множествами, с одной стороны, и операциями образования составных высказываний — с другой.

Если рассматривается несколько высказываний, то сопоставить каждому из этих высказываний некоторое множество можно вполне логичным путем. Сначала мы образуем множество всех логических возможностей для рассматриваемых высказываний и назовем его универсальным множеством. Затем каждому высказыванию мы поставим в соответствие подмножество тех логических возможностей универсального множества, для которых это высказывание истинно.

Определение. Пусть $X, Y, Z, \ldots$ означают некоторые высказывания, и пусть U — их множество логических возможностей. Пусть $A, B, C, \ldots$ означают подмножества U, для которых истинны соответственно высказывания $X, Y, Z, \ldots$. Тогда $A, B, C, \ldots$ называются соответственно множествами истинности высказываний $X, Y, Z, \ldots$.

Множество истинности двух высказываний X и Y показаны на диаграмме Эйлера–Венна. Здесь отмечены различные логические возможности этих двух высказываний (см. рис. 2.5).

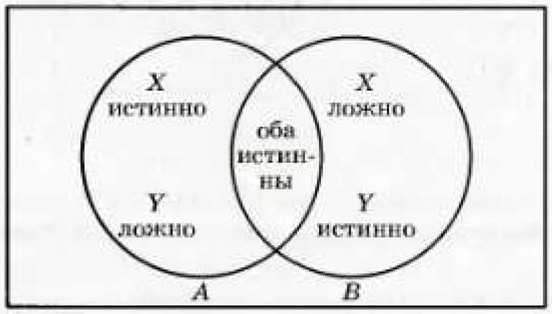


Рис. 2.5

Связь между высказыванием и его множеством истинности создает возможность «перевода» любой задачи, относящейся к составным высказываниям, в задачу теории множеств.

13. Соотношение между высказываниями и соответствующими им множествами истинности.

Мы рассмотрели такие множества истинности составных высказываний, которые образованы посредством связок $\vee$, $\wedge$, $\overline{}$. Все остальные связки можно определить через эти три основные и тем самым вывести, какие множества истинности им соответствуют. Например, известно, что импликация $X \to Y$ эквивалентна дизъюнкции $X \vee Y$. Поэтому множество истинности для $X \to Y$ будет тем же, что и множество истинности для $\overline{X} \vee Y$, т. е. оно будет иметь вид $\overline{A} \cup B$.

На диаграмме Эйлера–Венна выделенная область показывает множество истинности этого высказывания (см. рис. 2.6).

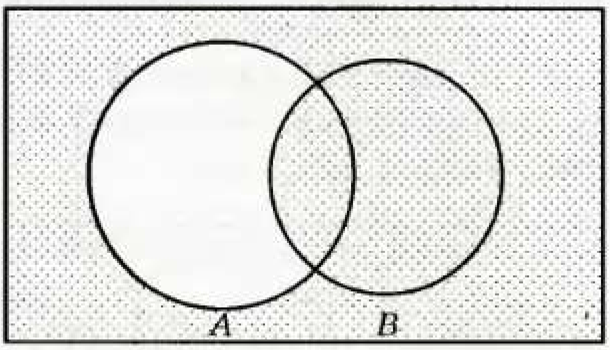


Рис. 2.6

Отметим, что незаштрихованная область на этой диаграмме показывает множество $A \setminus B = A \cap \overline{B}$, представляющее собой множество истинности высказывания $X \wedge \overline{Y}$. Поэтому заштрихованная область будет множеством $\overline{A \setminus B} = \overline{A \cap \overline{B}}$, которое является множеством истинности высказывания $\overline{X \wedge \overline{Y}}$. Таким образом, мы установили, что высказывания $X \to Y$, $\overline{X} \vee Y$, $\overline{X \wedge \overline{Y}}$ эквивалентны. Вообще, два высказывания эквивалентны тогда и только тогда, когда они имеют одни и те же множества истинности.

Рассмотрим отношение следствия. Напомним, что из X следует Y тогда и только тогда, когда импликация $X \to Y$ логически истинна. Но высказывание $X \to Y$ тогда и только тогда логически истинно, когда его множество истинности совпадает с U, т. е. $\overline{(A \setminus B)} = U$ и $A \setminus B = \varnothing$. Но если $A \setminus B$ пусто, то B включает в себя A. Отношение включения обозначается, как мы отмечали, $A \subset B$ и читается «A является подмножеством B». Таким образом, высказывание $X \to Y$ логически истинно тогда и только тогда, когда $A \subset B$.

Каждому высказыванию соответствует его множество истинности, каждой логической связке соответствует операция над множеством. Каждому отношению между высказываниями соответствует отношение между множествами истинности. Множествами истинности высказываний

$$X \vee Y; \quad X \wedge Y; \quad \overline{X} \quad \text{и} \quad X \to Y$$

служат соответственно:

$$A \cup B; \quad A \cap B; \quad \overline{A} \quad \text{и} \quad \overline{A \setminus B}.$$

Высказывание X логически истинно, если $A = U$, и логически ложно, если $A = \varnothing$. Высказывание X и Y эквивалентны тогда и только тогда, когда $A = B$; из X следует Y тогда и только тогда, когда $A \subset B$.

14. Абстрактные законы операций над множествами.

Эти законы очень напоминают элементарные законы алгебры высказываний.

По этой причине множество, его подмножества и законы сочетания подмножеств образуют алгебраическую систему, называемую **булевой алгеброй**. Система составных высказываний, подчиняющаяся таким законам, тоже называется **булевой алгеброй**. Таким образом, любую из этих систем можно изучать или с алгебраической, или с логической точки зрения.

Ниже перечислены основные законы, действующие в булевых алгебрах.

Законы для объединения и пересечения:

1. $A \cup A = A$	**7.** $A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$
2. $A \cap A = A$	**8.** $A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$
3. $A \cup B = B \cup A$	**9.** $A \cup U = U$
4. $A \cap B = B \cap A$	**10.** $A \cap \varnothing = \varnothing$
5. $A \cup (B \cup C) = (A \cup B) \cup C$	**11.** $A \cap U = A$
6. $A \cap (B \cap C) = (A \cap B) \cap C$	**12.** $A \cup \varnothing = A$

Законы для дополнений:

1. $\overline{\overline{A}} = A$	**4.** $\overline{(A \cup B)} = \overline{A} \cap \overline{B}$
2. $A \cup \overline{A} = U$	**5.** $\overline{(A \cap B)} = \overline{A} \cup \overline{B}$
3. $A \cap \overline{A} = \varnothing$	**6.** $\overline{U} = \varnothing$

Законы для разностей множеств:

1. $A \setminus B = A \cap \overline{B}$	**6.** $A \setminus A = \varnothing$
2. $U \setminus A = \overline{A}$	**7.** $((A \setminus B) \setminus C) = A \setminus (B \cup C)$
3. $A \setminus U = \varnothing$	**8.** $A \setminus (B \setminus C) = (A \setminus B) \cup (A \cap C)$
4. $A \setminus \varnothing = A$	**9.** $A \cup (B \setminus C) = (A \cup B) \setminus (C \setminus A)$
5. $\varnothing \setminus A = \varnothing$	**10.** $A \cap (B \setminus C) = (A \cap B) \setminus (A \cap C)$

Доказательство каждого из перечисленных законов основано на определении равенства множеств и определений операций над множествами. Напомним, что множество A равно множеству B, если они состоят из одних и тех же элементов или оба пусты. Докажем один из законов для дополнений: $\overline{A \cup B} = \overline{A} \cap \overline{B}$.

Пусть $x \in \overline{A \cup B}$. По определению операции дополнения это означает, что $x \notin A \cup B$, но $x \in U$. Следовательно, $x \notin A$ и одновременно $x \notin B$. Таким образом, $x \in \overline{A}$ и $x \in \overline{B}$. Из определения операции пересечения получаем, что $x \in \overline{A} \cap \overline{B}$. Поэтому, учитывая произвольность элемента $x \in \overline{A \cup B}$, имеем $\overline{A \cup B} \subset \overline{A} \cap \overline{B}$.

Пусть теперь $x \in \overline{A} \cap \overline{B}$. Это значит, что $x \in \overline{A}$ и $x \in \overline{B}$. Таким образом, $x \notin A$ и $x \notin B$. Поэтому $x \notin A \cup B$. Следовательно, $x \in U \setminus (A \cup B) = \overline{(A \cup B)}$. Поскольку x — произвольный элемент из $\overline{A} \cap \overline{B}$, то окончательно получаем $\overline{A} \cap \overline{B} \subset \overline{A \cup B}$.

Приходим к выводу, что $\overline{A} \cap \overline{B} = \overline{A \cup B}$.

15. Кортежи и декартово произведение множеств. Степень множества.

Определение. Пусть даны множества $X_1, X_2, \dots, X_n$. Кортежем длины n, составленным из элементов этих множеств, называется конечная последовательность $\alpha = \langle x_1, x_2, \dots, x_n \rangle$, где для всех k, $1 \leqslant k \leqslant n$, имеем $x_k \in X_k$.

Элемент x_k называется k-й координатой или k-й компонентой кортежа α.

Два кортежа равны в том и только в том случае, когда они имеют одинаковую длину, причем их координаты, стоящие на местах с одинаковыми номерами, равны, т. е. кортежи $\alpha = \langle x_1, \dots, x_m \rangle$ и $\beta = \langle y_1, \dots, y_n \rangle$ равны только в том случае, когда $m = n$, причем $x_k = y_k$ для всех $1 \leqslant k \leqslant n$.

Кортежи длины два называют упорядоченными парами, длины три — упорядоченными тройками, ..., длины n — упорядоченными n-ками. Для краткости речи слово «упорядоченные» часто опускают.

Кортеж, не содержащий ни одной координаты, т. е. кортеж длины 0, называется пустым.

Основные отличия понятий кортежа и множества следующие:

а) в множестве порядок элементов не играет роли, а кортежи, отличающиеся порядком элементов, различны, даже в случае, когда они имеют одинаковый состав;

б) в множестве все элементы различны, а в кортеже координаты могут повторяться.

В дальнейшем, чтобы различать множества и кортежи, будем элементы множества заключать в фигурные скобки, а координаты кортежа — в угловые.

Пусть $A_1, A_2, \dots, A_n$ — некоторые множества. Их декартовым произведением называют множество, состоящее из кортежей вида $\langle a_1, a_2, \dots, a_n \rangle$, где $a_1 \in A_1$; $a_2 \in A_2$; ...; $a_n \in A_n$. Декартово произведение обозначается так: $A_1 \times A_2 \times \dots \times A_n$.

Произведение

$$\underbrace{A \times A \times \dots \times A}_{n \text{ раз}}$$

сокращенно обозначается как A^n и называется декартовой n-й степенью множества A.

16. Бинарные отношения в множестве и их свойства.

Пусть A и B два конечных множества. Декартовым произведением множеств A и B называют множество $A \times B$, состоящее из всех упорядоченных пар $\langle a, b\rangle$, где $a \in A$, $b \in B$.

Бинарным отношением между элементами множеств A и B называется любое подмножество R множества $A \times B$, т. е. $R \subset A \times B$.

По определению, бинарным отношением называется множество пар. Если R — бинарное отношение (т. е. множество пар), то говорят, что параметры x и y связаны бинарным отношением R, если пара $\langle x, y\rangle$ является элементом R, т. е. $\langle x, y\rangle \in R$.

Высказывание: «предметы x и y связаны бинарным отношением R» записывают в виде $x\,R\,y$.

Таким образом, $x\,R\,y \leftrightarrow \langle x, y\rangle \in R$.

Program Files\Skillbrains\lightshot

Если $R \subset A \times A$, то говорят, что бинарное отношение определено на множестве A.

Областью определения бинарного отношения R называется множество, состоящее из таких x, для которых $\langle x, y\rangle \in R$ хотя бы для одного y.

Область определения бинарного отношения будем обозначать δ_R.

17. Отношения строгого и нестрогого порядка.

Исследование бинарных отношений на рефлексивность, симметричность и транзитивность; выделение классов эквивалентности.

Определение 1. Отношение называется отношением нестрогого порядка, если оно является рефлексивным, антисимметричным и транзитивным.

Определение 2. Отношение называется отношением строгого порядка, если оно является антирефлексивным, антисимметричным и транзитивным.

Оба типа отношений вместе называются отношениями порядка. Элементы сравнимы по отношению порядка, если выполняется одно из двух отношений. Множество, на котором задано отношение порядка, называется полностью упорядоченным, если любые два его элемента сравнимы. В противном случае, множество называется частично упорядоченным.

Определение. Отношение называется *отношением эквивалентности* (или просто *эквивалентностью*), если оно рефлексивно, симметрично и транзитивно. Эта система обладает следующими свойствами:

1) она образует *разбиение* множества M, то есть классы попарно не пересекаются;
2) любые два элемента из одного класса эквивалентны;
3) любые два элемента из разных классов не эквивалентны.

Все эти свойства прямо следуют из определения отношения эквивалентности.

18. Отображение множеств. Функции.

Соответствие f, сопоставляющее каждому элементу x множеств X один и только один элемент множества Y, называется **отображением** множества X во множество Y.

Элемент множества Y, соответствующий при отображении f элементу x из X, обозначают $f(x)$ и называют **образом** элемента x при этом отображении.

Если $f(x) = y$, то элемент x называют **прообразом** элемента y при отображении f.

Совокупность всех прообразов элемента y при отображении f называется **полным прообразом** этого элемента и обозначается $f^{-1}(y)$, т. е. $f^{-1} = \{x : f(x) = y\}$.

Правая часть читается: «совокупность таких x, что $f(x) = y$».

Каждому подмножеству A множества X ($A \subset X$) соответствует его образ $f(A)$ при отображении f. Этот образ состоит из всех элементов множества Y, которые являются образами какого-нибудь элемента из A: $f(A) = \{y : y = f(a),\ a \in A\}$.

Каждому подмножеству B множества Y ($B \subset Y$) соответствует его полный прообраз $f^{-1}(B)$ при отображении f. Он состоит из всех элементов, образы которых принадлежат B: $f^{-1}(B) = \{x : f(x) \in B\}$.

Множество A называется областью определения отображения f, а множество $f(A)$ называется множеством значений этого отображения.

Частный случай отображения множества X в множество Y имеет место, если каждый элемент множества Y имеет прообраз. В этом случае отображение f называется **сюръективным**.

Если для каждого элемента $y \in Y$ существует не более одного прообраза, т. е. $f(x_1) \neq f(x_2)$ для любых $x_1, x_2 \in X$, если $x_1 \neq x_2$, то отображение f называется **инъективным**.

Если отображение f сюръективно и инъективно, то оно называется **биективным** (взаимно-однозначным).

С точки зрения отображений два множества называются количественно-эквивалентными, если между ними можно установить биективное отображение.

Если между элементами множеств A и B существует биективное отображение, то множества A и B называются равномощными.

Для конечных множеств A и B понятие равномощности означает, что они имеют одно и то же число элементов. Таким образом, если A — конечное множество, содержащее n элементов, то **мощностью множества** A называется число n и обозначается $|A|$, т. е. $|A| = n$.

Очевидно, что отношение равномощности является отношением эквивалентности, поэтому **равномощные** множества часто называют **эквивалентными**.

19. Определенность и неопределенность функций. Композиция отображений.

В основе всех разделов дискретной математики лежит понятие функции.

Функцией называется бинарное отношение f, если из $\langle x, y\rangle \in f$ и $\langle x, z\rangle \in f$ следует, что $y = z$.

Две функции f и g равны, если они состоят из одних и тех же элементов. Область определения функции и область ее значений задается так же, как и для бинарных отношений. Если область определения $\delta_f = X$ и область значений $\rho_f \subset Y$, то говорят, что функция f задана на множестве X со значениями во множестве Y, при этом f отображает множество X во множество Y. Это отображение обозначается как $f : X \to Y$.

Если f — функция, то вместо $\langle x, y\rangle \in f$ пишут $y = f(x)$ и говорят, что y — значение, соответствующее аргументу x, или y — образ элемента x при отображении f. При этом x называют прообразом элемента y.

Назовем f — n-местной функцией из X в Y, если $f : X^n \to Y$. Тогда записываем $y = f(x_1, \ldots, x_b)$ и говорим, что y — значение функции при значении аргументов $x_1, \ldots, x_n$.

Пусть $f : X \to Y$. Функция f называется инъективной, если для любых x_1, x_2, y из $y = f(x_1)$ и $y = f(x_2)$ следует, что $x_1 = x_2$.

Функция f называется сюръективной, если для любого элемента $y \in Y$ существует элемент $x \in X$ такой, что $y = f(x)$.

Функция f называется биективной, если f одновременно сюръективна и инъективна.

Если существует биективная функция $f : X \to Y$, то говорят, что f осуществляет взаимно-однозначное соответствие между множествами X и Y.

Если $f : X \to Y$, а $g : Y \to Z$, то функция $F : X \to Z$, определенная для каждого $x \in X$ формулой $F(x) = g(f(x))$, называется композицией (суперпозицией) функции f и g, или **сложной функцией**.

Пусть задана функция $f : X \to Y$ и ρ_f — множество ее значений. Совокупность всевозможных упорядоченных пар вида $\langle y, f^{-1}(y)\rangle$, $y \in \rho_f$ образует функцию, которая называется **обратной функцией** для функции f и обозначается f^{-1}.

Обратная функция f^{-1} ставит в соответствие каждому элементу $y \in \rho_f$ его прообраз $f^{-1}(y)$, т. е. некоторое множество элементов. Заметим, что для того, чтобы f^{-1} являлась функцией, достаточно, чтобы функция f была инъективной.

20. Метод математической индукции. База индукции. Индукционный переход. Полная и неполная индукция.

Метод математической индукции - не просто распространенный метод решения олимпиадных задач, но и способ доказательства многих утверждений в математической науке. *База индукции*: первое утверждение ряда верно.
Переход индукции: если верно какое-то утверждение ряда (неважно, какое именно*),* то верно и следующее за ним утверждение ряда.
Решение задач на выполнение операций в алгебре вычетов
Для степени $y=2^n$(n–натуральное число) установить классы сравнимости. Установить зависимость последней цифры этой степени от ее показателя.

Решение и комментарии. Как известно, натуральные степени числа 2 оканчиваются цифрами {2, 4, 8, 6}. См. таблицу нескольких степеней числа 2. Определим функцию, которая ставит в соответствие каждому натуральному числу *n* последнюю цифру числа 2^n:

$$f: N \to \{2, 4, 8, 6\},$$

Эта функция f(n) периодична с периодом 4. Это значит, что для целого числа k: f(n)=f(n+4)= f(n+4k),.

Причем справедливы так же равенства: f(n)=f(n-4)= f(n-4k)

Последнее равенство означают, что для любого *n* нужно найти минимальное натуральное *m,* такое, что f(m) = f(m + 4k) = f(n).

Но это задача на делении с остатком числа nна 4:

n=4k+m, k-частное, *m - остаток*.

Очевидно, последняя цифра числа 2" зависит от остатка, полученного при делении показателя n степени 2^n на 4.

Отразим этот факт в записи функции: f(n)= f (nmod 4)

Из этой формулы можно установить, если f (nmod 4) = 0, то $m \vdots 4$

n	2^n	Последняя цифра $2^т$
1	2	2
2	4	4
3	8	8
4	16	6
5	32	2
6	64	4
7	128	8
8	256	6

При делении чисел на 4 $\forall n \in N$, останки могут быть: 0,1,2,3. Таким образом, в частности, множество всех возможных показателей степени 2^n для любого n состоит из четырех подмножеств: *4k, 4k*+ 1, *4k+ 2, 4k+3.*

21. Основные правила комбинаторики. Методы алгоритмического перечисления (генерации) основных комбинаторных объектов: перестановка, сочетание, размещение.

Пусть необходимо выполнить последовательно k действий. Если первое действие можно выполнить n_1 способами, второе n_2 способами и так далее до k — действия, которое можно выполнить n_k способами, то все k действий можно выполнить $n_1 \cdot n_2 \cdot \ldots \cdot n_k$ способами.

Другим часто применяемым в комбинаторике правилом является правило суммы. Это правило формулируется следующим образом: если элемент x может быть выбран m способами, а элемент y — другими n способами, то выбор «либо x либо y» может быть осуществлен $m + n$ способами.

Пусть A — конечное множество, состоящее из n элементов $|A| = n$.

а) *Размещения.* Кортежи длины k $(1 \leqslant k \leqslant n)$, состоящие из различных элементов n-элементного множества A (кортежи отличаются один от другого как самими элементами, так и их порядком), называются размещениями из n элементов множества A по k. Число таких размещений будем обозначать A_n^k (буква A от французского слова *arrangement* — размещение). Схема выбора состоит в выборе k элементов из n-элементного множества без возвращения.

Тогда необходимо совершить k действий, причем первое действие можно совершить n способами, второе $(n-1)$ способами и т. д., k-е действие $n-(k-1)$ способами. Согласно комбинаторному правилу умножения, получим формулу: $A_n^k = n \cdot (n-1) \cdot \ldots \cdot (n-k+1)$.

Если умножить и разделить полученное выражение на $(n-k)!$, получим:

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

где $n! = n \cdot (n-1) \cdot \ldots \cdot 3 \cdot 2 \cdot 1$ и называется факториалом числа n (читается n-факториал). Причем: $0! = 1$; $1! = 1$; $2! = 1 \cdot 2 = 2$; $3! = 1 \cdot 2 \cdot 3 = 6$; $4! = 1 \cdot 2 \cdot 3 \cdot 4 = 24$; $5! = 4! \cdot 5 = 120\ldots$.

Пусть, например, дано множество $A\{1, 3, 5\}$. Выпишем все размещения из трех элементов по два: $\langle 1, 3\rangle$, $\langle 1, 5\rangle$, $\langle 3, 5\rangle$, $\langle 3, 1\rangle$, $\langle 5, 3\rangle$, $\langle 5, 1\rangle$.

Число этих размещений можно найти по формуле

$$A_3^2 = \frac{3!}{(3-2)!} = \frac{1 \cdot 2 \cdot 3}{1!} = 6.$$

б) *Перестановки.* Пусть у нас есть n-элементное множество A, будем строить из этого множества размещения в виде кортежей длины n. Эти размещения будут отличаться друг от друга только порядком, поскольку в каждом из них встречаются по одному разу все элементы множества A. Такие размещения называются перестановками и обозначаются P_n (буква P от английского слова *permutation* — перестановка). Поскольку $P_n = A_n^n$, то число перестановок вычисляется по формуле $P_n = n!$.

в) *Сочетания.* Из n-элементного множества A будем строить упорядоченные множества длины k ($1 \leqslant k \leqslant n$), не учитывая порядок элементов, т. е. размещения с одними и теми же элементами, расположенными в разном порядке, будем считать равными.

Такие размещения называются сочетаниями и обозначаются C_n^k (буква C от английского слова *combination* — комбинация).

Число сочетаний из n элементов по k меньше числа размещений из n элементов по k в P_k раз, т. е. $C_n^k = \dfrac{A_n^k}{P_k}$.

Используя это утверждение, выведем формулу для вычисления числа сочетаний:

$$C_n^k = \frac{A_n^k}{P_k} = \frac{n!}{(n-k)! \cdot k!} = \frac{n(n-1)\dots(n-k+1)}{k!}.$$

Из этой формулы непосредственно вытекает, что $C_0^0 = C_n^0 = C_n^n = 1$; $C_n^1 = n$; $C_n^k = C_n^{n-k}$.

22. Комбинация элементов с повторениями.

Размещениями с повторениями из n элементов по k называются кортежи длины k, составленные из n — элементного множества A. Число этих кортежей обозначают $\overline{A_n^k}$. Черта указывает на возможность повторения элементов $\overline{A_n^k} = n^k$. Например, сколько пятизначных номеров можно составить из элементов множества $\{1, 2, 3, 4, 5, 6, 7, 8, 9\}$? Такими номерами являются кортежи длины 5, составленные из девятиэлементного множества, где схема выбора состоит в выборе 5 элементов из девятиэлементного множества с возвращением, т. е. для каждого из пяти элементов есть девять способов выбора, т. е. $\overline{A_9^5} = 9^5 = 59\,049$.

Перестановкой с повторениями состава $(n_1, \dots, n_k)$ из букв $(a_1, \dots, a_k)$ называют любой кортеж длины $n = n_1 + n_2 + \dots + n_k$, в который a_1 входит n_1 раз, a_2 входит n_2 раз, ..., a_k — n_k раз. Число таких перестановок обозначают

$$P(n_1, n_2, \dots, n_k) = \frac{n!}{n_1!n_2!\dots n_k!} = \frac{(n_1 + n_2 + \dots + n_k)!}{n_1!n_2!\dots n_k!}.$$

Сочетания с повторениями. Пусть имеются предметы n видов и из них составляется набор, содержащий k элементов, т. е. различными исходами будут всевозможные наборы длины k, отличающиеся составом, и при этом отдельные наборы могут содержать повторяющиеся элементы. Такие наборы называются сочетаниями с повторениями, а их общее число определяется формулой : $\overline{C_n^k} = C_{n+k-1}^k$.

С числами C_n^k связано функциональное тождество, называемое формулой бинома Ньютона. Из элементарной математики хорошо известны формулы сокращенного умножения:

$$(a+b)^2 = a^2 + 2ab + b^2; \quad (a+b)^3 = a^3 + 3a^2b + 3ab^2 + b^3.$$

Эти формулы можно записать так:

$$(a+b)^2 = C_2^0 a^2 b^0 + C_2^1 ab + C_2^2 a^0 b^2;$$
$$(a+b)^3 = C_3^0 a^3 b^0 + C_3^1 a^2 b^1 + C_3^2 ab^2 + C_3^3 a^0 b^3.$$

Имеет место и общая закономерность: справедливо равенство:

$$(a+b)^n = C_n^0 a^n b^0 + C_n^1 a^{n-1} b + C_n^2 a^{n-2} b^2 + \ldots + C_n^n a^0 b^n.$$

Это равенство и называется биномом Ньютона, а коэффициенты $C_n^0, C_n^1, C_n^2, \ldots, C_n^n$ называются биномиальными коэффициентами.

Если положить $a = b = 1$, то из формулы бинома Ньютона вытекает следующее важное соотношение: $(1+1)^n = C_n^0 + C_n^1 + C_n^2 + \ldots + C_n^n = 2^n$ — формула суммы биномиальных коэффициентов.

Если положить в биноме Ньютона $a = 1$, $b = -1$, то

$$C_n^0 - C_n^1 + C_n^2 - \ldots + (-1)^n C_n^n = 0.$$

Поскольку $C_n^k = C_n^{n-k}$, то биномиальные коэффициенты, равноотстоящие от концов в формуле бинома Ньютона, равны.

23. Предикаты. Применение предикатов в алгебре.

Рассмотрим пример: «x простое число». Это выражение не является высказыванием, но если в нем переменную x заменить на определенное число, то получим высказывание. Причем при замене x на число 3 получим истинное высказывание, тогда как при замене x на 8 получим ложное высказывание.

Таким образом, выражение: «x простое число» можно рассматривать как функцию $P(x)$, зависящую от переменной x. Область определения $P(x)$ — множество чисел, а область значения — высказывание.

Определение. Предикат — функция, значениями которой являются высказывания о n объектах, представляющих значения аргументов.

Чтобы задать n-местный предикат $P(x_1, x_2, \ldots, x_n)$, следует указать множества $X_1, X_2, \ldots, X_n$ — области изменения переменных $x_1, x_2, \ldots, x_n$, причем чаще всего рассматривается случай, когда $X_1 = X_2 = \ldots = X_n$.

С теоретико-множественной точки зрения предикат определяется заданием подмножества M в декартовом произведении $X_1 \times X_2 \times \ldots \times X_n$.

Переменные $x_1, x_2, \ldots, x_n$ называются *предметными переменными*. Элементы множеств $X_1, X_2, \ldots, X_n$ называются *предметами*. Множество M — множество кортежей длины n $\langle x_1, x_2, \ldots, x_n \rangle$ называется *полем предиката* $P(x_1, x_2, \ldots, x_n)$.

Будем обозначать предметные переменные малыми буквами конца латинского алфавита (иногда будем снабжать эти буквы индексами) $x, y, z, \ldots, x_1, x_2, \ldots, x_n$.

Предметы из множеств $X_1, X_2, \ldots, X_n$ — малыми буквами начала латинского алфавита $a, b, c, \ldots, a_1, a_2, a_3 \ldots$

Предикаты — большими буквами латинского алфавита с приписанными предметными переменными или без них $A(x, x)$, B, $F(x, y)$, $P(x_1, \ldots, x_n)$.

Число переменных будем указывать как верхний индекс у предиката: $P^k(x_1, x_2, \ldots, x_k)$ — k местный предикат, $Q^2(x, y)$ — двуместный предикат, $P(x)$ — одноместный предикат.

Итак, k-местный предикат — $P^k(x_1, x_2, \ldots, x_k)$ есть функция, предметные переменные которой принимают значения из некоторого множества M_k, а сама она принимает только два значения: истина (1) или ложь (0), т. е.

$$P^k(x_1, x_2, \ldots, x_k) : \; M_k \to \{1, 0\}.$$

Например, если X — множество действительных чисел, то $x^2 > 1$ — одноместный предикат.

Если X, Y — множества действительных чисел, то $xy = 5$ — двуместный предикат.

Предикат называется *разрешимым*, если существуют такие кортежи, компоненты которых обращают предикат в истинное высказывание.

Если предикат при подстановке любых конкретных элементов из соответствующих множеств обращается в истинное высказывание, он называется *тождественно истинным*.

Если предикат при подстановке любых конкретных элементов из соответствующих множеств обращается в ложное высказывание, он называется *тождественно ложным*.

К предикатам, определенным на одном и том же множестве, можно применять операции алгебры высказываний: конъюнкцию, дизъюнкцию, импликацию, эквивалентность, отрицание и получать новые предикаты.

Например, если к предикатам «$x = y$» и «$x < y$» — обозначим их соответственно $P(x, y)$ и $Q(x, y)$ — применить операцию конъюнкции, то получим новый предикат $P(x, y) \wedge Q(x, y)$.

24. Булева алгебра предикатов.

Так как к предикатам можно применять логические операции, то для них справедливы основные законы булевой алгебры.

1. $\overline{\overline{P}} = P$

2. $P \vee Q = Q \vee P$

3. $P \wedge Q = Q \wedge P$

4. $P \vee (Q \vee R) = (P \vee Q) \vee R$

5. $P \wedge (Q \wedge R) = (P \wedge Q) \wedge R$

6. $P \vee (Q \wedge R) = (P \vee Q) \wedge (P \vee R)$

7. $P \wedge (Q \vee R) = (P \wedge Q) \vee (R \wedge R)$

8. $\overline{P \vee Q} = \overline{P} \wedge \overline{Q}$; $\overline{P \wedge Q} = \overline{P} \vee \overline{Q}$

9. $P \vee P = P$; $P \wedge P = P$

10. $P \vee 1 = 1$; $P \wedge 0 = 0$; $P \vee 0 = P$; $P \wedge 1 = P$;
$P \vee \overline{P} = 1$; $P \wedge \overline{P} = 0$

11. $P \vee (P \wedge Q) = P$

12. $P \wedge (P \vee Q) = P$

25. Кванторы. Формулы логики предикатов.

Помимо операций алгебры высказываний, в логике предикатов есть две операции, которые связаны с природой предикатов. Пусть дан предикат $P(x)$, зависящий от одной переменной и определенный на поле M.

а) Выражение $(\forall x)P(x)$ означает высказывание, истинное только в том случае, когда предикат $P(x)$ истинен для всех предметов из поля M. Выражение $(\forall x)P(x)$ читается «для всякого x, $P(x)$», здесь символ $\forall$ — квантор общности.

б) Выражение $(\exists x)P(x)$ означает высказывание, истинное только в том случае, когда предикат $P(x)$ истинен хотя бы для одного предмета из поля M. Выражение $(\exists x)P(x)$ читается «существует x, что $P(x)$»; символ $\exists$ — квантор существования.

Рассмотрим примеры применения операций квантирования к предикатам. Пусть даны предикаты над полем натуральных чисел:

1) $x^2 = x \cdot x$, тогда $(\forall x)(x^2 = x \cdot x)$ — истинное высказывание;

2) $x + 2 = 7$, тогда $(\forall x)(x + 2 = 7)$ — ложное высказывание; а $(\exists x)(x + 2 = 7)$ — истинное высказывание;

3) $x + 2 = x$, тогда $(\forall x)(x + 2 = x)$ — ложное высказывание.

Название	Прочтение	Обозначение
Квантор общности	«все», «всякий», «каждый», «любой»	$\forall$
Квантор существования	«Хотя бы один», «найдется», «существует»	$\exists$

переменные x и z — связанные, а y и v — свободные.

Если квантор общности или квантор существования применяется не к одноместному предикату, а к какому-нибудь k-местному предикату, то в результате этого получается снова предикат, но за счет связывания одной предметной переменной получаемый предикат будет $(k-1)$-местным.

26. Равносильные формулы логики предикатов. Приведенные и нормальные формы в логике предикатов.

Применяя к переменным предикатам операции $\wedge$; $\vee$; $\rightarrow$; $\leftrightarrow$; $\overline{}$; $\forall$; $\exists$, получим формулы логики предикатов, т. е. *формулой логики предикатов* называется выражение, составленное из переменных предикатов с помощью логических операций и кванторов и обращающееся в конкретный предикат при подстановке вместо переменных конкретных предикатов.

Например, $((\forall x)W(x, y) \vee B) \rightarrow U(z)$ — формула логики предикатов.

Формула логики предикатов называется *тавтологией*, если при подстановке любых конкретных предикатов она всегда обращается в тождественно истинный предикат.

(1) Формула логики предикатов называется *атомарной*, т. е. *элементарной*, если в ней нет связанных переменных.

(2) Пусть F — формула, тогда $\overline{F}$ — тоже формула. Свободные и связанные переменные формулы $\overline{F}$ — это соответственно свободные и связанные переменные формулы F.

(3) Пусть F и G — формулы, причем в них нет предметных переменных, которые были бы связаны в одной формуле и свободны в другой.

Тогда $F \wedge G$, $F \vee G$, $F \to G$, $F \leftrightarrow G$ — формулы, в которых свободные переменные формул F и G остаются свободными, а связанные — связанными.

(4) Пусть F — формула, содержащая свободную переменную x. Тогда $(\forall x)F$, $(\exists x)F$ — тоже формулы, в которых переменная x связана, а остальные свободные переменные, входящие в F, остаются свободными.

Заметим, что по определению формулы никакая переменная не может быть одновременно свободной и связанной.

Значение формулы определено лишь тогда, когда задана какая-то *интерпретация* входящих в нее символов. Под интерпретацией понимают систему $M = \langle M, f \rangle$, состоящую из непустого множества M и соответствия f, которое сопоставляет каждой формуле определенный предикат. При заданной интерпретации предметные переменные пробегают множество M, а логические символы и символы кванторов имеют свой обычный смысл.

Пусть формулы F и G имеют одно и то же множество свободных переменных (в частности, пустое). Формулы F и G *равносильны в данной интерпретации*, если они принимают одинаковые значения на любом наборе свободных переменных, т. е. выражают в данной интерпретации один и тот же предикат.

Формулы F и G равносильны *на множестве* M, если они принимают одинаковые значения во всех интерпретациях заданных на множестве M.

Формулы F и G равносильны в логике предикатов, если они равносильны на всех множествах ($F \equiv G$).

Рассмотрим правила перехода от одних формул к другим, им равносильным.

(1) *Перенос квантора через отрицание.* Пусть $W(X)$ — формула, содержащая свободную переменную x. Тогда справедливы равносильности:

$$\overline{(\forall x)W(x)} \equiv (\exists x)\overline{W(x)}, \quad \overline{(\forall x)\overline{W(x)}} \equiv (\exists x)W(x),$$

$$\overline{(\exists x)W(x)} \equiv (\forall x)\overline{W(x)}, \quad \overline{(\exists x)\overline{W(x)}} \equiv (\forall x)W(x).$$

(2) *Вынос квантора за скобки.* Пусть формула $W(x)$ содержит свободную переменную x, а формула B не содержит переменной x. Формулы $W(x)$ и B удовлетворяют третьему правилу создания формул. Тогда справедливы равносильности:

$$(\exists x)(W(x) \wedge B) \equiv (\exists x)W(x) \wedge B, \quad (\forall x)(W(x) \wedge B) \equiv (\forall x)W(x) \wedge B,$$
$$(\exists x)(W(x) \vee B) \equiv (\exists x)W(x) \vee B, \quad (\forall x)(W(x) \vee B) \equiv (\forall x)W(x) \vee B.$$

(3) *Перестановка одноименных кванторов.* Имеем

$$(\exists x)(\exists y)W(x, y) \equiv (\exists y)(\exists x)W(x, y), \quad (\forall x)(\forall y)W(x, y) \equiv (\forall y)(\forall x)W(x, y).$$

(4) *Переименование связанных переменных.* Заменяя связанную переменную формулы W другой переменной, не входящей в эту формулу, в кванторе и всюду в области действия квантора, получим формулу, равносильную W.

Формулы, в которых из логических символов имеются только символы конъюнкция, дизъюнкция и отрицание, причем символ отрицания встречается над символами предикатов, будем называть *приведенными*.

Например, формула $(\forall x_1)A_1^{(1)}(x_1) \vee (\exists x_2)\overline{A_2^{(2)}(x_2, x_3)}$ — приведенная; формула $\overline{(\forall x_2)}A_1^{(1)}(x_2) \to A_2^{(1)}(x_1)$ — неприведенная.

Для любой формулы существует равносильная ей приведенная формула, причем множества свободных и связанных переменных этих формул совпадают.

Такая приведенная формула называется *приведенной формой* данной формулы.

В логике высказываний мы ввели две нормальные формы — дизъюнктивную нормальную форму и конъюнктивную нормальную форму.

В логике предикатов также имеется нормальная форма, цель которой — упрощение процедуры доказательств.

Приведенная формула называется *нормальной*, если она не содержит символов кванторов или все символы кванторов стоят впереди (т. е. логические символы и символы предикатов стоят в области действия каждого квантора).

Для любой приведенной формулы существует равносильная ей нормальная формула той же длины (под длиной формулы будем понимать общее число входящих в нее символов предикатов, логических символов и символов кванторов).

Нормальная формула называется *нормальной формой* данной формулы.

Алгоритм преобразования формул в нормальную форму:

1. Исключить логические связки $\leftrightarrow$ и $\rightarrow$ с помощью формул

$$F \leftrightarrow G = (F \rightarrow G) \wedge (G \rightarrow F), \quad F \rightarrow G = \overline{F} \vee G.$$

2. Использовать закон $\overline{\overline{F}} = F$, законы де Моргана:

$$\overline{(F \vee G)} = \overline{F} \wedge \overline{G}, \quad \overline{(F \wedge G)} = \overline{F} \vee \overline{G},$$

законы

$$\overline{((\forall x)F(x))} = (\exists x)\overline{(F(x))}, \quad \overline{((\exists x)F(x))} = (\forall x)\overline{(F(x))},$$

чтобы пронести знак отрицания внутрь формулы.

3. Переименовать связанные переменные, если это необходимо.

4. Использовать равносильные формулы логики предикатов, чтобы вынести кванторы в самое начало формулы для приведения ее к нормальной форме. Например, приведем формулу $(\forall x)P(x) \rightarrow (\exists x)Q(x)$ к нормальной форме:

$$(\forall x)P(x) \rightarrow (\exists x)Q(x) = \overline{((\forall x)P(x))} \vee (\exists x)Q(x) = (\exists x)\overline{(P(x))} \vee (\exists x)Q(x) =$$
$$= (\exists x)\left(\overline{P(x)} \vee Q(x)\right).$$

Следовательно, нормальная форма формулы $(\forall x)P(x) \rightarrow (\exists x)Q(x)$ — это $(\exists x)(\overline{P(x)} \vee Q(x))$.

27. Исчисление предикатов.

Исчисление предикатов называют еще теорией первого порядка.

В исчислении предикатов, так же как и в исчислении высказываний, на первом по важности месте стоит проблема *разрешимости*.

Но в исчислении высказываний проблема разрешимости состояла в решении вопроса является ли данная сложная функция тождественно истинной, выполнимой или тождественно ложной?

Теперь же вопрос следует поставить иначе. Принимает ли данная функция значение 1 при:

а) любых предметных переменных и любых предикатах,

б) на некотором множестве предметных переменных и любых предикатах,

в) при некоторых значениях предметных переменных и при некоторых предикатах,

г) является ли она тождественно ложной, т. е. невыполнимой?

Таким образом, в логике предикатов, в отличие от логики высказываний, нет эффективного способа для распознавания общезначимости функций.

Поэтому в исчислении предикатов указывается некоторая совокупность формул, которые называются аксиомами и составляют *аксиоматическую теорию*, и указывается конечное множество отношений между формулами, составляющее *правила вывода*.

Аксиоматическая теория и правила вывода и составляют исчисления предикатов.

Символами исчисления предикатов или алфавитом исчисления предикатов являются символы предметных переменных, символы предикатов, логические символы (отрицание и импликация), символы кванторов, а также скобки и запятая.

Аксиомы исчисления предиката.

Пусть A, B и C — любые формулы.

Аксиома 1. $A \to (B \to A)$.

Аксиома 2. $(A \to (B \to C)) \to ((A \to B) \to (A \to C))$.

Аксиома 3. $(\overline{B} \to \overline{A}) \to \left((\overline{B} \to A) \to B\right)$.

Аксиома 4. $(\forall x_i)A(x_i) \to A(x_j)$, где формула $A(x_i)$ не содержит переменной x_j.

Аксиома 5. $A(x_i) \to (\exists x_j)A(x_j)$, где формула $A(x_i)$ не содержит переменной x_j.

Правила вывода исчисления предикатов.

(1) Пусть $(A(x) \to B)$ и B не содержит переменной x, тогда

$$\left(((\exists x)A(x)) \to B\right).$$

Это правило связывания квантором существования.

(2) Пусть $B \to A(x)$ и B не содержит переменной x, тогда

$$\left(B \to ((\forall x)A(x))\right).$$

Это правило связывания квантором общности.

(3) Связанную переменную формулы B можно заменить другой переменной, не являющейся свободной в B. Это правило переименования связанной переменной.

28. Основные понятия теории графов. Степень вершины. Маршрут, цепи, циклы. Связность графа.

Во многих прикладных задачах изучаются системы связей между различными объектами. Объекты называются вершинами и отмечаются точками или кружочками, а связи между вершинами — отрезками, соединяющими пары точек, и эти отрезки называются ребрами. Рассмотрение таких систем и приводит к понятию графа.

Граф представляет собой непустое конечное множество вершин V и множество ребер E, оба конца которых принадлежат множеству V. Обозначать граф будем $G(V, E) = \langle V, E\rangle$, $V \neq \varnothing$, $E \subset V \times V$.

При изображении графов на рисунках или схемах ребра могут быть прямолинейными или криволинейными; длины ребер и расположение вершин произвольны.

Вершины, которые не принадлежат ни одному ребру, называются *изолированными*.

Обозначать вершины будем буквами $v_1, v_2, v_3, \ldots, v_n$, т. е. $V = \{v_1, v_2, \ldots, v_n\}$.

Пусть v_1, v_2 — вершины, $e = \langle v_1, v_2\rangle$ — соединяющее их ребро. Тогда вершина v_1 и ребро e *инцидентны*. Вершина v_2 и ребро e также *инцидентны*. Два ребра, инцидентные одной вершине, называются *смежными*. Две вершины, инцидентные одному ребру, также называются *смежными*.

Число вершин графа G обозначим p, а число ребер обозначим q

$$p : p(G) = |V|; \quad q : q(G) = |E|.$$

Граф называется полным, если каждые две различные вершины его соединены одним и только одним ребром.

Степенью вершины называется число ребер графа, которым принадлежит эта вершина. Степень графа еще называют его *валентностью* и обозначают $d(v)$. Вершина графа, для которой $d(v) = 0$, является *изолированной*, для которой $d(v) = 1$ — *висячей*.

Вершина называется *нечетной*, если $d(v)$ — нечетное число. Вершина называется *четной*, если $d(v)$ — четное число. Степень каждой вершины полного графа на единицу меньше числа его вершин.

В графе $G(V, E)$ сумма степеней всех его вершин — число четное, равное удвоенному числу ребер графа. Число нечетных вершин любого графа четно. Во всяком графе с n вершинами, где $n \geqslant 2$, всегда найдутся, по меньшей мере, две вершины с одинаковыми степенями.

Если в графе с n вершинами ($n \geqslant 2$) в точности две вершины имеют одинаковую степень, то в этом графе всегда найдется либо в точности одна вершина степени 0, либо в точности одна вершина степени $n - 1$.

Маршрутом в графе называется чередующаяся последовательность вершин и ребер, в которой любые два соседних элемента инцидентны:

$$v_0, e_1, v_1, e_2, v_2, \ldots, e_k, v_k.$$

Если $v_0 = v_k$, то маршрут замкнут, в противном случае открыт.

Если все ребра различны, то маршрут называется цепью. Если все вершины (а значит, и ребра) различны, то маршрут называется простой цепью. В цепи $v_0, e_1, v_1, e_2, v_2, \ldots, e_k, v_k$ вершины v_0 и v_k называются концами цепи, т. е. цепь концами v_0 и v_k соединяет вершины v_0 и v_k. Цепь, соединяющая вершины v_0 и v_k, обозначается $\langle v_0, v_k \rangle$. Очевидно, что если есть цепь, соединяющая вершины v_0 и v_k, то есть и простая цепь, соединяющая эти вершины. Замкнутая цепь называется *циклом*; замкнутая простая цепь называется *простым циклом*. Число циклов в графе $G(V, E)$ обозначается $z(G)$. Граф без циклов называется *ациклическим*.

Длиной маршрута называется количество ребер в нем (с повторениями). Если маршрут $M = v_0, e_1, v_1, e_2, v_2, \ldots, e_k, v_k$, то длина маршрута M равна k, обозначается $|M| = k$.

Две вершины графа называются связными, если существует соединяющая их простая цепь. В противном случае две вершины называются несвязными.

Граф называется связным, если каждые две вершины его связные. Граф называется несвязным, если хотя бы две его вершины несвязные.

29. Ориентированные графы.

Если элементы множества E графа $G(V, E)$ — упорядоченные пары, то граф называется *ориентированным* или *орграфом*.

Ребро e графа G называется ориентированным, если одну вершину считают началом ребра, а другую — концом, на рисунке его изображают стрелкой между вершинами. Таким образом, граф, все ребра которого ориентированы, называется ориентированным графом.

Одна и та же вершина ориентированного графа может служить началом для одних ребер и концом для других, поэтому различают две степени вершины: степень выхода и степень входа.

Степенью выхода вершины орграфа называется число *выходящих* из вершины ребер.

Степенью входа вершины орграфа называется число *входящих* в вершину ребер.

В орграфах в зависимости от сочетания степеней входа и выхода для данной вершины рассматриваются три случая.

Изолированной вершиной называется вершина, у которой и степень входа и степень выхода равны 0.

Источником называется вершина, степень выхода которой положительна, а степень входа равна 0.

30. Изоморфизм графов.

Два графа $G_1(V_1, E_1)$ и $G_2(V_2, E_2)$ называются изоморфными, если между множествами их вершин существует биективное (взаимнооднозначное) соответствие, такое, что вершины соединены ребрами в одном из графов в том и только в том случае, когда соответствующие им вершины соединены в другом графе. Если ребра графа ориентированы, то их направление в изоморфных графах должно совпадать. Изоморфизм графов есть отношение эквивалентности, так как обладает свойствами рефлексивности, симметричности, транзитивности. Для того чтобы граф G_1 был изоморфен графу G_2, необходимо и достаточно существования такой подстановки, которая бы установила взаимнооднозначное соответствие между вершинами графа, а также между их ребрами.

При замене графа любым ему изоморфным все свойства графа сохраняются. Строго говоря, графы, отличающиеся только нумерацией вершин, являются изоморфными.

Алгоритм распознавания изоморфизма двух графов $G_1(X; E)$ и $G_2(Y; E)$.

1. Подсчитываем число вершин каждого графа (число вершин должно совпадать, в противном случае графы неизоморфные).

2. Выписываем все элементы обоих графов в естественной упорядоченности и определяем пары $(x_i; x_j)$ и $(y_i; y_j)$ для каждого элемента, где x_i; y_i — число исходов для каждой вершины графов G_1 и G_2, а x_j; y_j — число заходов для соответствующих графов.

3. Для каждого элемента x графа G_1 ищем такой элемент y графа G_2, что выполняется условие: число исходов x совпадает с числом исходов y, и число заходов x совпадает с числом заходов y. Найденные элементы x и y соединяем ребром, т. е. строим граф соответствия (если соответствия нет, то графы не изоморфны).

4. Выписываем подстановку, которая переводит граф G_1 в граф G_2.

31. Плоские графы. Операции над графами.

Граф $G(V, E)$ называется плоским, если на плоскости его можно изобразить так, что все пересечения его ребер являются вершинами графа $G(V, E)$.

В качестве характеристики плоского представления графа вводится понятие грани.

Гранью в плоском представлении графа называется часть плоскости, ограниченная простым циклом и не содержащая внутри других циклов.

Рассмотрим графы $G_1(V_1, E_1)$ и $G_2(V_2, E_2)$.

а) *Дополнением* графа $G_1(V_1, E_1)$ называется граф $\overline{G_1}(V_1, \overline{E_1})$, множеством вершин которого является множество V_1, а множеством его ребер является множество $\overline{E_1} = \{e \in V_1 \times V_1 : e \notin E_1\}$.

б) *Объединением* графов $G_1(V_1, E_1)$ и $G_2(V_2, E_2)$ при условии, что $V_1 \cap V_2 = \varnothing$; $E_1 \cap E_2 = \varnothing$, называется граф $G_1(V_1, E_1) \cup G_2(V_2, E_2)$, множеством вершин которого является множество $V_1 \cup V_2$, а множеством его ребер является множество $E_1 \cup E_2$.

в) *Пересечением* графов $G_1(V_1, E_1)$ и $G_2(V_2, E_2)$ называется граф $G_1(V_1, E_1) \cap G_2(V_2, E_2)$, множеством вершин которого является множество $V_1 \cap V_2$, а множеством его ребер — множество $E_1 \cap E_2$.

г) *Суммой по модулю два* графов $G_1(V_1, E_1)$ и $G_2(V_2, E_2)$ при условии, что $V_1 \cap V_2 = \varnothing$; $E_1 \cap E_2 = \varnothing$, называется граф $G_1(V_1, E_1) \oplus G_2(V_2, E_2)$, множеством вершин которого является множество $V_1 \cup V_2$, а множеством его ребер — множество $E_1 \oplus E_2$. Другими словами, этот граф не имеет изолированных вершин и состоит только из ребер, присутствующих либо в первом графе, либо во втором графе, но не в обоих графах одновременно.

Легко убедиться в том, что операции: объединение, пересечение и сумма по модулю два обладают свойством *коммутативности*.

32. Способы задания графов. Некоторые типы графов.

Аналитический способ задания графов

Граф $G(V, E)$ задан, если задано множество элементов V и отображение E множества V в V. Отображение E может быть как однозначным, так и многозначным. В общем случае на V и E никаких ограничений не накладывается.

Пусть дано множество $V = \{v_1, v_2, \ldots, v_n\}$, которое имеет мощность $|V| = n$. Вместо $V = \{v_1, v_2, \ldots, v_n\}$ иногда пишут

$$V = \{v_i\}, \quad i \in \{1, 2, \ldots, n\}.$$

Для того чтобы задать отображение E на V или, что то же самое, отображение V в V, необходимо каждому элементу $v_i \in V$ поставить в соответствие некоторое подмножество множества V, которому соответствует отображение E. Это подмножество обозначают через E_{v_i}. Поэтому $E_{v_i} \subset V$. Совокупность двух объектов: множества V и отображение E на V задает некоторый граф.

Другой формой аналитического способа задания является задание графа как совокупности множества элементов V и подмножества множества упорядоченных пар $\langle v_i, v_j \rangle \in V \times V$. Подмножество множества пар $\langle v_i, v_j \rangle$ декартова произведения $V \times V$ эквивалентно бинарному отношению R, заданному на множестве V. Поэтому множество V и бинарное отношение R на множестве V также определяет некоторый граф G.

Геометрический способ задания графов

Множество элементов V графа G изображают кружками, это множество вершин. Каждую вершину $v_i \in V$ соединяют линиями с теми вершинами $v_i \in V$, для которых выполняется условие $v_i \in E_{v_i}$. Множество линий, которое соответствует множеству упорядоченных пар $\langle v_i, v_j \rangle$, есть множество ребер графа.

Матричный способ задания графов

Квадратная матрица

$$\begin{Vmatrix} a_{11} & a_{12} & \ldots & a_{1n} \\ a_{21} & a_{22} & \ldots & a_{2n} \\ \ldots & \ldots & \ldots & \ldots \\ a_{n1} & a_{n2} & \ldots & a_{nn} \end{Vmatrix},$$

элементами которой являются нули и единицы, а также некоторое число m, называется *матрицей смежности* графа $G(V, E)$ тогда и только тогда, когда ее элементы образуются по следующему правилу: элемент $a_{i,j}$, стоящий на пересечении v_i-й строки и v_j-го столбца, равен единице, если имеется ребро, идущее из вершины v_i в вершину v_j, и $a_{i,j}$ равен нулю в противном случае. Элемент $a_{i,i}$ равен единице, если при вершине v_i имеется петля, и равен нулю в противном случае. Элемент $a_{i,j}$ равен некоторому числу m, где m — число ребер графа, идущее из вершины v_i в вершину v_j.

Пусть $v_i, \ldots, v_n$ — вершины, а $e_1, \ldots, e_m$ — ребра некоторого ориентированного графа $G(V, E)$. Матрица размером $(m \times n)$, где

$$a_{i,j} = \begin{cases} 1, \text{ если } e_j \text{ исходит из } v_i; \\ -1, \text{ если } e_j \text{ заходит в } v_i; \\ 0, \text{ если } e_j \text{ не инцидентна } v_i \end{cases}$$

называется матрицей инцидентности для ребер ориентированного графа.

Эйлеровы графы

К задачам на эйлеровы графы относятся головоломки, в которых требуется вычертить на плоскости одним росчерком замкнутые кривые, обводя каждый участок в точности один раз. Введем следующие понятия.

Эйлеровым путем в графе называется путь, содержащий все ребра графа.

Эйлеровым циклом или *эйлеровой цепью* называется цикл, содержащий все ребра графа и притом по одному разу.

Граф, обладающий эйлеровым циклом, называется эйлеровым графом.

Замкнутую линию, если ее можно начертить, не отрывая карандаша от бумаги, проходя при этом каждый участок в точности один раз, принято называть уникурсальной.

Рисунок графа, обладающего эйлеровым путем или эйлеровым циклом, является уникурсальной линией.

Теорема 1. Если граф $G(V, E)$ обладает эйлеровым циклом, то он связный и все его вершины четные.

Теорема 2. Если граф $G(V, E)$ связный и все его вершины четные, то он обладает эйлеровым циклом.

Для построения эйлерова цикла в связном графе со всеми вершинами четной степени применяется следующий алгоритм:

1. Выйти из произвольной вершины v_i. Каждое пройденное ребро зачеркнуть. Если путь l_1 замыкается в v_i и проходит через все ребра графа, то получим искомый эйлеров цикл.

2. Если остались непройденные ребра, то должна существовать вершина v_2, принадлежащая l_1 и ребру, не вошедшему в l_1.

3. Так как v_2 — четная, то число ребер, которым принадлежит v_2 и которые не вошли в путь l_1, тоже четно. Начнем новый путь l_2 из v_2 и используем только ребра, не принадлежащие l_1. Этот путь кончится в v_2.

4. Объединим теперь оба цикла: из v_i пройдем по пути l_1 к v_2, затем по l_2 и, вернувшись в v_2, пройдем по оставшейся части l_1 обратно в v_i.

5. Если снова найдутся ребра, которые не вошли в путь, то найдем новые циклы. Так как число ребер и вершин конечно, то процесс закончится.

Таким образом, замкнутую фигуру, в которой все вершины четные, можно начертить одним росчерком без повторений и начиная с любой точки.

На практике эйлеровым графом может быть план выставки; это позволяет расставить указатели маршрута, чтобы посетитель смог пройти по каждому залу в точности по одному разу.

Эйлеровы и гамильтоновы пути сходны по способу задания. Первые содержат все ребра, и притом по одному разу, вторые — все вершины по одному разу. Но, несмотря на внешнее сходство, задачи их отыскания резко отличаются по степени трудности. Для решения вопроса о существовании эйлерова цикла в графе достаточно выяснить, все ли его вершины четные.

Критерий же существования гамильтонова цикла на произвольном графе еще не найден.

Однако есть несколько достаточных условий существования гамильтоновых циклов в графе:

1. Всякий полный граф является гамильтоновым, так как он содержит простой цикл, которому принадлежат все вершины данного графа.

2. Если граф, помимо простого цикла, проходящего через все его вершины, содержит и другие ребра, то он также является гамильтоновым.

3. Если граф имеет один гамильтонов цикл, то он может иметь и другие гамильтоновы циклы.

33. Сети. Сетевые модели представления информации. Применение графов и сетей.

Граф называется **взвешенным** или **сетью**, если каждому его ребру поставлено в соответствие некоторое число (**вес**). Взвешенными графами могут быть схемы в электронике, электрические схемы, карты автомобильных и железных дорог и др. Например, на картах автодорог вершины являются населенными пунктами, ребра — дорогами, а весом — числа, равные расстоянию между населенными пунктами.

В строительстве сетевые графы применяются для наглядного изображения некоторого комплекса работ или производственных процессов. Ребрам графа могут соответствовать числа, означающие длину, уклон, запланированное время и другие характеристики.

Например, последовательность работ для монтажа каркаса здания изображена в виде графа (рис. 2.20).

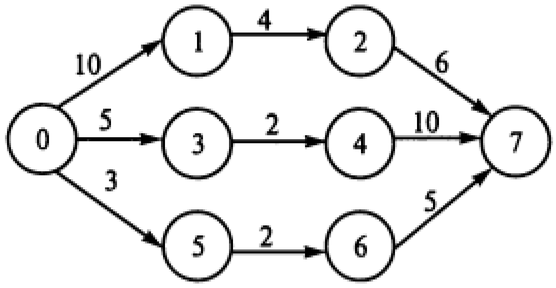
Числами обозначены технологические операции:

1 — рытье котлована;
2 — монтаж фундамента;
3 — завоз металлоконструкций;
4 — монтаж подъемного крана;
5 — монтаж каркаса здания.

В основе процесса планирования лежит некоторый сценарий, представляющий собой сеть, состоящую из вершин — пошагового описания действий и дуг — отношений между ними. Такой граф дает возможность, сравнивая альтернативы, планировать действия для достижения поставленной цели.

Сети широко используются в качестве моделей для представления знаний в интеллектуальных системах. Сетевая модель представления информации основана на том, что любые знания можно рассматривать как множества объектов (понятий) и связи между ними (отношения).

Понятия-объекты и другие элементы предметной области могут быть графически изображены в виде вершин, а отношения между ними — в виде дуг, связывающих эти вершины. Такое графическое представление информации (знаний) в интеллектуальных системах носит название **семантических сетей**. Они являются универсальным средством для представления знаний в интеллектуальных системах. Понятия, входящие в сети, можно описать с помощью фрейма. **Фреймом** называется минимально возможное описание сущности некоторого явления, объекта, события или процесса. Состоит фрейм из набора стандартных единиц — *слотов*, содержащих определенный минимум информации о его содержании и назначении. Семантическая сеть в виде некоторой со-



вокупности фреймов нуждается в указании отношения между ее вершинами, что тоже возможно осуществить в виде слота. Семантические сети широко применяются в информатике, например, для операций **поиска по образцу**, где в виде сетей представляется база данных. Результат такого поиска можно изобразить графом. Используются сети и для графической иллюстрации системы отношений базы данных.

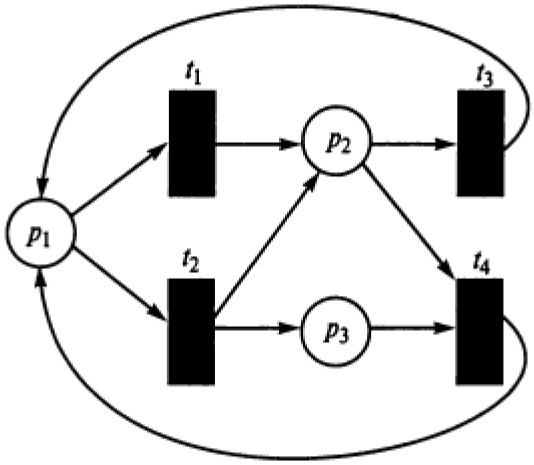


Рис. 2.22. Сеть Петри

Широко применяются сети для графического изображения различных логических схем в теории автоматов, например схемы с памятью, у которых каждый узел $F(i)$ — функция алгебры логики (см., например, рис. 4.17, 4.18).

Для формального описания совокупности процессов, протекающих одновременно, используют **сети Петри**. Они представляют собой ориентированные графы, состоящие из вершин двух видов: некоторых позиций и переходов, причем позиции изображают кружочками, а переходы — «планками» (рис. 2.22). Сети Петри предназначены для описания действия дискретных процессов во времени. Такие сети дают возможность моделировать ситуации протекания параллельных процессов, прослеживать возможные варианты их взаимодействия, выявлять нежелательные ситуации. Также в виде сетей изображаются схемы устройств, например радиоприемника или телевизора.

С помощью графов-деревьев решают задачи планирования (дерево целей, дерево переборов вариантов). Графы используют также для иллюстрации классификаций в различных областях знаний при построении **иерархических структур** сложных систем.

Структуру такого типа имеют, например, предприятия (его составные части: цехи, бригады, участки и т.д.), учебные заведения (колледж состоит из факультетов, которые, в свою очередь, состоят из курсов, курсы — из групп). Такой же зависимости подчинены армейские соединения (дивизия, полк, батальон, рота, взвод, отделение, отдельные солдаты). Аналогично устроена любая административно-территориальная структура: республика, области, районы, населенные пункты. По путям этих деревьев движутся информационные потоки: сверху вниз — распоряжения, руководящие указания, снизу вверх — отчеты о работе, оперативная информация. Так как путь от листа к корню единственный, то его можно использовать для опознания компонентов системы. Например, почтовый адрес представляет собой «путь в дереве», аналогичный административно-территориальному. В разделе «Кому» указывается страна, республика, область, район, населенный пункт, улица, дом, квартира. Аналогично классифицируют объекты в любой науке. Получаемая классификация служит примером **иерархической структуры.** Например, в биологии: класс, отряд, семейство, род, вид. Соответствующий граф содержит элементы разных уровней, корень — класс, а листья — отдельные виды животных.

Иногда связи между объектами образуют не дерево, но все же их можно представить в виде графа. Это бывает в тех случаях, когда, например, происходит подчинение не одному, а нескольким независимым службам (соподчинение между собой).

В информатике иерархические структуры применяют при описании базы данных, вычислительных сетей, сетей связи, организационных систем. С помощью графа можно графически изображать родословные (генеалогическое дерево или древо).

Бинарный поиск. Бинарные деревья применяются в информатике для одной из самых распространенных в прикладных науках операций — поиску. К нему прибегают, когда необходимо найти в некотором упорядоченном массиве (множестве) определенную информацию. Например, в телефонном справочнике — номер какого-нибудь абонента, в словаре — определенное слово, в файле — сведения о зарплате сотрудников некоторого предприятия, сведения о зарплате отдельного работника и т.д.

Последовательный, или линейный, поиск является наиболее общим и простым методом выявления интересующей информации: каждый элемент множества проверяется на соответствие заданным условиям. Если множество неупорядочено, то такой процесс будет носить случайный характер. Если множество было предварительно упорядочено (распределение по алфавиту фамилий в телефонном справочнике, слов в словаре, распределение служащих предприятия по табельным номерам в информации о зарплате), то удобнее использовать другой, более эффективный метод — бинарный поиск.

Бинарный поиск основан на методе половинного деления. Поиск начинается с середины множества. Если первый же элемент удовлетворяет условию, то процесс поиска завершен, если не удовлетворяет, то процесс продолжается в любой из половин. Если искомый элемент так и не найден, то анализируется вторая половина до тех пор, пока не обнаружится соответствие найденного элемента заданным условиям. Бинарные деревья необходимы, когда делается выбор одной альтернативы из двух.

Различные виды графов и деревьев находят широкое применение в учебном процессе. Поэтому первоначальные сведения о них необходимы для успешного обучения в различных областях, так как с их помощью часто передается учебная информация.

На процедуре бинарного поиска с использованием графа-дерева основана работа ЭВМ с базой данных. Информация о базе может быть представлена несколькими способами, в том числе матричным. Так называемые реляционные базы хранят различную информацию в форме таблиц, причем порядок строк и столбцов задается при вводе данных. В базах возможна длительная работа с информацией, ее реорганизация и обновление. С помощью автоматического поиска данных происходит их отбор на основании запроса по определенным характерным признакам.

Запросы в виде сложносоставных высказываний образуются из простейших с помощью *логических связок* и ряда *логических операций*. Все методы работы с информацией — поиск, обработка и накопление — основаны на широком применении законов *математической логики*, знание которой необходимо для понимания принципов работы вычислительных машин, различных электронных устройств.

34. Вычислимые функции и алгоритмы.

Определение. *Вычислимая функция* — это такая функция, для которой существует вычисляющий ее значения алгоритм, т. е. функция $y = f(x_1, x_2, \ldots, x_n)$ называется вычислимой, если существует алгоритм, позволяющий определить значения функции при любых значениях переменных $x_1, x_2, \ldots, x_n$.

Определение. *Алгоритмом* называется точное предписание, определяющее вычислительный процесс, который ведет от варьируемых исходных данных к искомому результату, т. е. алгоритм — это совокупность правил, определяющих данный вычислительный процесс.

Для каждого алгоритма существует некоторая совокупность возможных исходных данных — объектов, к которым имеет смысл применять рассматриваемый алгоритм. Для каждого алгоритма выделяется *область применимости*: если процесс применения алгоритма к какому-либо объекту заканчивается выдачей результатов, то говорят, что он применим к этому объекту.

Алгоритм задает функцию, определенную на его области применимости и ставящую в соответствие каждому элементу этой области результат применения к нему алгоритма.

Не все объекты, встречающиеся в математике, могут служить исходными данными, результатами или промежуточными данными алгоритма.

В алгоритмических процессах могут участвовать лишь *конструктивные объекты*.

Определение. *Конструктивными объектами* будем называть натуральные и рациональные числа, полиномы с натуральными или рациональными коэффициентами, матрицы с натуральными или рациональными элементами, слова в некотором алфавите и т. д., т. е. такие объекты, которые могут быть построены целиком и представлены для рассмотрения.

Поскольку возможными исходными данными и результатами алгоритма могут быть лишь конструктивные объекты, то лишь конструктивные объекты могут быть аргументами и значениями вычислимой функции.

Анализ известных в математике алгоритмов дал возможность выявить характерные его свойства.

Свойство 1. *Дискретность.* Алгоритм описывает процесс последовательного построения величин, идущий в дискретном времени. Необходимый для вычисления интервал времени разбит на малые отрезки — такты. Система величин в конце каждого такта получается в результате осуществления элементарного шага алгоритма (определенной программы преобразования) из системы величин, имеющейся к началу такта.

Свойство 2. *Детерминированность (определенность)* требуется, чтобы метод действия (вычисления) был настолько точен и общепонятен, чтобы не оставалось места произволу. Этот метод можно сообщить другому лицу в виде конечного числа указаний, т. е. программа преобразований в каждом такте однозначно определена.

Свойство 3. *Результативность.* Это свойство, называемое иногда еще направленностью алгоритма, требует, чтобы алгоритмическая процедура, примененная к любой задаче данного типа, через конечное число шагов останавливалась и после остановки можно было бы прочесть искомый результат.

Свойство 4. *Массовость.* Алгоритм служит не для решения какой-либо одной конкретной задачи, а для решения целого класса задач. Процедуру для решения одной индивидуальной задачи не называют алгоритмом.

На основе понятий вычислимой функции и алгоритма определяются понятия разрешимого предиката, разрешимого множества и перечислимого множества.

Определение. Предикат $P(x_1, x_2, \ldots, x_n)$, определенный на множестве целых чисел, называется алгоритмически разрешимым или просто разрешимым, если существует алгоритм для определения значения предиката P при любых значениях переменных $x_1, x_2, \ldots, x_n$.

Определение. Множество называется разрешимым, если существует алгоритм, распознающий принадлежность произвольного элемента к этому множеству.

Определение. Множество называется перечислимым, если оно есть множество значений какой-нибудь вычислимой функции, определенной на всем натуральном ряду.

35. Рекурсивные функции.

а) *Оператор суперпозиции (подстановки).* В основе этого оператора лежит способ вычисления, который состоит в том, что если мы умеем вычислять функции f и g, то мы сумеем вычислить функцию $h = f(g)$.

б) *Оператор примитивной рекурсии.* В основе этого оператора лежит широко распространенный способ вычисления функции натурального аргумента, состоящий в том, что задается $f(0)$ и так называемая рекуррентная формула $f(n + 1) = h(f(n))$, позволяющая вычислить $f(n + 1)$, если известно $f(n)$. С помощью этой формулы последовательно вычисляются $f(1)$, $f(2)$ и т. д. Такой процесс последовательного вычисления значений функции называется рекурсией.

Пусть заданы какие-либо числовые частичные функции: n-местная g и $(n + 2)$-местная h. $(n + 1)$-местная частичная функция f возникает из функций g и h с помощью оператора примитивной рекурсии, если она может быть задана *схемой примитивной рекурсии*:

$$\begin{cases} f(x_1, \ldots, x_n, 0) = g(x_1, \ldots, x_n), \\ f(x_1, \ldots, x_n, 1) = h(x_1, \ldots, x_n, 0, f(x_1, \ldots, x_n, 0)), \\ f(x_1, \ldots, x_n, 2) = h(x_1, \ldots, x_n, 1, f(x_1, \ldots, x_n, 1)), \\ \ldots \\ f(x_1, \ldots, x_n, y + 1) = h(x_1, \ldots, x_n, y, f(x_1, \ldots, x_n, y)). \end{cases}$$

Совокупность этих равенств для любых функций g и h однозначно определяет значения функции f. Таким образом, для каждых двух частичных числовых функций g от n переменных и h от $(n + 2)$ переменных существует одна и только одна функция f от $(n + 1)$ переменной, возникающая в результате примитивной рекурсии. Оператор примитивной рекурсии будем обозначать через $f = R(g, h)$.

Итак, схема примитивной рекурсии образует некоторый, индуктивный процесс построения функции h, при котором на нулевом шаге используется функция g, а на каждом последующем шаге — значение функции f от аргументов $x_1, \ldots, x_n$, номер y предыдущего шага и значение функции h, вычисленного на предыдущем шаге. Заметим, что оператор примитивной рекурсии можно применять по любым переменным, входящим в функции f, g и h, но всегда нужно указывать, по каким переменным этот оператор проводится.

в) *Оператор минимизации.* Существует третья важная операция, порождающая новые вычислимые функции, а именно неограниченная минимизация, или просто минимизация.

Пусть $f(x, y)$ — функция, и мы хотим определить функцию $g(x)$, положив $g(x)$ = наименьшее y такое, что $f(x, y) = 0$. При этом из вычислимости f должна следовать вычислимость g. Для этого введем следующее определение оператора минимизации μ, который дает вычислимые функции из вычислимых функций.

Для каждой функции $f(x, y)$

$$\mu f(x, y) = 0 = \begin{cases} \text{наименьший } y \text{, такой, что } f(x, y) \text{ определенно для всех} \\ z \leqslant y \text{ и } f(x, y) = 0 \text{, если такой } y \text{ существует,} \\ \text{не определено в противном случае.} \end{cases}$$

$\mu y(\ldots)$ читается как «наименьший y, такой, что...».

Этот оператор называют иногда μ-оператором.

Сформулируем следующую теорему: Пусть $f(x, y)$ вычислима; тогда вычислима и функция $g(x) = \mu y\,(f(x, y) = 0)$.

Определение. Функция f называется примитивно-рекурсивной, если существует последовательность функций $f_0, f_1, \ldots, f_n$, в которой $f_n = f$ и всякая функция f_i является либо простейшей (базисной), либо получается из предыдущих функций с помощью конечного числа применений оператора суперпозиции S или оператора примитивной рекурсии R (простейшие функции являются примитивно-рекурсивными).

Итак, примитивно-рекурсивные функции — это арифметические функции, которые сопоставляются по определенным правилам примитивно-рекурсивным описанием.

Всякая примитивно-рекурсивная функция является вычислимой, так как процесс ее определения с помощью примитивно-рекурсивного описания дает одновременно и способ вычисления значений функции для любых числовых значений ее аргументов.

Определение. Функция $f(x_1, \ldots, x_n)$ называется частично-рекурсивной, если она может быть получена из простейших примитивно-рекурсивных функций с помощью конечного числа применений операторов суперпозиции, примитивной рекурсии и минимизации.

Для каждой частично-рекурсивной функции f существует вычислительный процесс, посредством которого любое натуральное число x перерабатывается в значение $f(x)$ функции f. Этот процесс не дает определенного результата тогда и только тогда, когда значение функции f в точке x не определено.

Определение. Пусть $P(x_1, \ldots, x_n)$ — n-местный предикат (зависит от n переменных) на множестве натуральных чисел N_0. Функция $\varphi_p(x_1, \ldots, x_n)$ называется характеристической функцией для предиката P, если эта функция удовлетворяет условию

$$\varphi_P(x_1, \ldots, x_n) = \begin{cases} 0, & \text{если } P(x_1, \ldots, x_n) \text{ истинно;} \\ 1, & \text{если } P(x_1, \ldots, x_n) \text{ ложно.} \end{cases}$$

Определение. Предикат $P(x_1, \ldots, x_n)$ разрешим, если характеристическая функция $\varphi(x_1, \ldots, x_n)$ вычислима; предикат $P(x_1, \ldots, x_n)$ неразрешим, если функция $\varphi_p(x_1, \ldots, x_n)$ невычислима.

36. Нормальный алгоритм Маркова.

Определение. Алфавитом называется любая конечная система различных символов.

ие: C:\Program Files\Skillbrains\lightshot ами называются символы, составляющие алфавит.

Определение. Словом в алфавите называется любая конечная последовательность букв в этом алфавите.

Определение. Пустым словом называется слово, не содержащее ни одной буквы и обозначаемое символом $\wedge$.

Примеры.

1) {а, ъ, ?, 7, $*$} — алфавит; а, ъ, ?, 7, $*$ — буквы.

2) $\{a, b, c\}$ — алфавит; ac, a, $abbca$, $bbbbb$, $bbacab$ — слова этого алфавита.

3) Алфавит исчисления высказываний состоит из букв A, B, Q, R, P и других, возможно с индексами, логических связок — $\wedge$, $\vee$, $\to$, а также вспомогательных символов (,).

Основная операция на словах — операция приписывания слова к слову: если дано слово, имеющее вид $a_0a_1\ldots a_n$, и слово вида $b_0b_1\ldots b_m$, то можно образовать новое слово $a_0a_1\ldots a_nb_0b_1\ldots b_m$, полученное приписыванием или соединением слов.

Если $\wedge$ — пустое слово, а a — слово, то $\wedge a = a\wedge = a$.

Определение. Два конкретных слова $a_0a_1\ldots a_n$ и $b_0b_1\ldots b_m$ и в алфавите A равны, т. е. $a_0a_1\ldots a_n = b_0b_1\ldots b_m$, если $n = m$ и $a_1 = b_1, \ldots, a_n = b_n$. Все пустые слова считаются равными.

Определение. Если $a_0a_1\ldots a_n$ — слово, состоящее из n букв, где $a_1\ldots a_n \in A$, то n называется длиной этого слова. Длиной пустого слова будет число 0.

Рассмотрим два слова: слово P и слово Q в некотором алфавите A. Если слово P является частью слова Q, то говорят, что слово P входит в слово Q.

Определение. Ассоциативным исчислением называется совокупность всех слов в данном алфавите A вместе с конечной системой допустимых подставок, т. е. чтобы задать ассоциативное исчисление достаточно задать алфавит и систему подстановок.

Любую подстановку $L - M$ можно применять к некоторому слову R этого алфавита следующим способом: если в слове R имеется одно или несколько включений слова L, то любое из этих включений может быть заменено словом M и наоборот, если имеется включение слова M, то его можно заменить словом L.

К полученным с помощью допустимых подставок словам можно снова применить допустимые подстановки: так будут получены новые слова.

Определение. Два слова P_1 и P_2 в некотором ассоциативном исчислении называются смежными, если одно из них может быть преобразовано в другое при помощи однократного применения некоторой допустимой подстановки.

Определение. Последовательность слов $P_1, P_2, P_3, \ldots, Q$ называется дедуктивной цепочкой, ведущей от слова P к слову Q, если каждые из двух рядом стоящих слов этой цепочки являются смежными.

Определение. Два слова P и Q называются эквивалентными, если существует дедуктивная цепочка, ведущая от слова P к слову Q. Отношение эквивалентности обозначается $P \leftrightarrow Q$. Очевидно, что если $P \leftrightarrow Q$, то, поскольку допустимые подстановки можно применять в обе стороны, $Q \leftrightarrow P$.

Иногда рассматривается специальный вид ассоциативного исчисления, которое задается алфавитом и системой ориентированных подстановок вида $P \rightarrow Q$. Это означает, что подстановку разрешается проводить лишь слева направо, т. е. заменять вхождение слова P на слово Q, но не наоборот.

Ясно, что в таком ассоциативном исчислении из эквивалентности $P \leftrightarrow Q$ не следует, что $Q \leftrightarrow P$.

А. А. Марковым было дано точное математическое определение *нормального алгоритма*.

Задается алфавит A и фиксируется схема подстановок. Алгоритм предписывает, исходя из произвольного слова P в алфавите A, просмотреть формулы подстановок в том порядке, в каком они заданы в схеме, разыскивая формулу с левой частью, входящей в P. Если такой формулы не найдется, то процесс обрывается. В противном случае берется первая из таких формул и делается подстановка ее правой части вместо первого вхождения ее левой части в P, что дает новое слово P_1 в алфавите A. После выполнения первого шага приступают ко второму шагу, отличающемуся от первого только тем, что роль P играет P_1. Далее делают аналогичный третий шаг и т. д. до тех пор, пока не придется оборвать процесс. Оборваться же он может лишь двумя способами: во-первых, когда мы получим такое слово P_n, что ни одна из левых частей формул схемы подстановок не будет в него входить; во-вторых, когда при получении слова P_n нам придется применить последнюю формулу. В обоих случаях мы считаем, что наш алгоритм перерабатывает слово P в слово P_n.

Различные нормальные алгоритмы отличаются друг от друга лишь алфавитами и системами допустимых подстановок. Чтобы задать какой-либо нормальный алгоритм достаточно задать алфавит и систему подстановок.

Понятие алгоритма в некотором алфавите было уточнено Марковым следующим образом: всякий алгоритм в алфавите A эквивалентен нормальному алгоритму в этом же алфавите.

37. Машины Тьюринга.

Тьюрингскими алгоритмами являются алгоритмы, конструктивные объекты которых можно кодировать в виде слов в некотором алфавите.

Пусть заданы два конечных алфавита A и Q. A назовем внешним алфавитом; Q назовем внутренним алфавитом (или алфавитом состояний).

Внешний алфавит — это конечный алфавит символов, которые подаются на вход машины Тьюринга и выдаются на ее выходе.

Внутренний алфавит — это конечный алфавит символов, комбинации которых определяют внутреннее состояние машины Тьюринга. Элементы A называются внешними символами машины, элементы Q — внутренними состояниями или просто состояниями.

Предположим, что символы $\to$, R, L не входят ни в A, ни в Q.

Командой назовем слово одного из следующих трех видов: $qa \to rb$; $qa \to rbR$; $qa \to rbL$, где $q, r \in Q$; $a, b \in A$. Команды, подобно формулам языка, можно читать по-русски. Команда первого вида читается так: «находясь в состоянии q и наблюдая букву a, следует перейти в состояние r и напечатать букву b». Команда второго вида: «находясь в состоянии q и наблюдая букву b, следует перейти в состояние r, напечатать букву b и затем передвинуться вправо». Команда третьего вида читается так же, как и команда второго вида, за исключением концовки: «...и затем передвинуться влево».

Конечная последовательность команд называется *программой* машины.

Пусть теперь фиксированы алфавиты A и Q, а также две буквы $q_0, q_1 \in Q$ и одна буква $a_0 \in A$. Мы называем q_0 — начальным состоянием, q_1 — финальным или заключительным состоянием, букву a_0 называем бланком, или пустой клеткой, или пробелом.

Алфавит $Q \times A$ назовем алфавитом наблюдаемых букв: про букву $\langle q, a\rangle \in Q \times A$ говорим, что a наблюдается в состоянии q.

Определение. Машиной Тьюринга называется набор $M = \langle A, Q, \Pi, q_0, q_1, a_0 \rangle$, где A — внешний алфавит, Q — внутренний алфавит, Π — программа машины, q_0 — начальное состояние, q_1 — заключительное состояние, a_0 — бланк или пробел.

Мы говорим, что машина M переводит машинное слово K_1 в K_2 и имеем $M : K_1 \xrightarrow{M} K_2$, если

1) существует команда k_i программы Π машины M такая, что

$$k_i : K_1 \xrightarrow{M} K_2;$$

2) для всех $j < i$ не существует машинного слова K_3 такого, что $k_i : K_1 \xrightarrow{M} K_3$.

Таким образом, если $M : K_1 \xrightarrow{M} K_2$, то машинное слово K_2 определяется однозначно по M и K_1. Конечно, вполне может оказаться, что некоторое машинное слово K_1 не переводится машиной M ни в какое машинное слово K_2.

Протокол вычислений машины Тьюринга M есть конечная последовательность машинных слов $K_0, K_1, \ldots, K_n$ такая, что:

1) K_0 — начальное машинное слово;

2) $M : K_i \xrightarrow{M} K_{i+1}$;

3) если машинное слово K_i входит в протокол $M : K_i \xrightarrow{M} K_{i+1}$, причем K_i — не заключительное машинное слово, то K_{i+1} также входит в протокол вычислений. Т. е. протокол не оканчивается словом K_i, если вычисления можно продолжить дальше;

4) протокол вычислений может содержать не более одного заключительного слова, и если протокол действительно содержит заключительное машинное слово, то этот протокол конечен и заключительным машинным словом является последний его член.

Машину Тьюринга можно представить себе как определенный тип вычислительной машины, обрабатывающую слова в алфавитах и порождающую функцию, которая перерабатывает некоторые слова в алфавите A в слова алфавита A. То есть если $\alpha, \beta \in A^*$, то $M(\alpha) = \beta$.

Заметим, что функция M может быть определена не на всех словах алфавита A. Машина M определена на слове α, и пишем $M(\alpha)$, если M определена на начальном слове.

38. Понятия конечного автомата. Определения и способы задания конечного автомата.

Теория автоматов представляет собой раздел дискретной математики, изучающий модели преобразователей дискретной информации. Такими преобразователями являются как реальные устройства (компьютеры, живые организмы), так и воображаемые устройства (аксиоматические теории, математические машины). По сути конечный автомат можно охарактеризовать как устройство M, имеющее входной и выходной каналы при этом в каждый из дискретных моментов времени, называемых тактовыми моментами, оно находится в одном из конечных состояний.

По входному каналу в каждый момент времени $t = 1, 2, \dots$ в устройство M поступают входные сигналы (из некоторого конечного множества сигналов). Задается закон изменения состояния к следующему моменту времени в зависимости от входного сигнала и состояния устройства в текущий момент времени. Выходной сигнал зависит от состояния и входного сигнала в текущий момент времени (рис. 7.1).

Конечный автомат является математической моделью реальных дискретных устройств по переработке информации.

Конечным автоматом называется система $A = (X; Q; Y; \varphi; \psi)$, где $X; Q; Y$ — произвольные непустые конечные множества.

Множество $X = \{a_1, \dots, a_m\}$ называется входным алфавитом, а его элементы — входными сигналами, их последовательности — входными словами. Множество $Q = \{q_1, \dots, q_n\}$ называется множеством состояний автомата, а его элементы — состояниями. Множество $Y = \{b_1, \dots, b_p\}$ называется выходным алфавитом, его элементы — выходными сигналами, их последовательности — выходными словами.

Функция $\varphi : X \times Q \to Q$ называется функцией переходов. Функция $\psi : X \times Q \to Y$ называется функцией выходов, т. е. $\varphi(x, q) \in Q$; $\psi(x, q) \in Y$ для $\forall x \in X$; $\forall q \in Q$.

С конечным автоматом ассоциируется воображаемое устройство, которое работает следующим образом. Оно может находиться в состоянии из множества Q, воспринимать сигналы из множества X и выдавать сигналы из множества Y.

Табличное задание автомата

Из определения автомата следует, что его всегда можно задать таблицей с двумя входами, содержащей m строк и n столбцов, где на пересечении столбца q и строки a стоят значения функций $\varphi(a_i; q_j)$; $\psi(a_i; q_j)$.

$a \setminus q$	q_1	...	q_j	...	q_n
a_1	$\varphi(a_1; q_1)$; $\psi(a_1; q_1)$	...	$\varphi(a_1; q_j)$; $\psi(a_1; q_j)$	...	$\varphi(a_1; q_n)$; $\psi(a_1; q_n)$
...	...	...	...	...	...
a_i	$\varphi(a_i; q_1)$; $\psi(a_i; q_1)$	...	$\varphi(a_i; q_j)$; $\psi(a_i; q_j)$	...	$\varphi(a_i; q_n)$; $\psi(a_i; q_n)$
...	...	...	...	...	...
a_m	$\varphi(a_m; q_1)$; $\psi(a_m; q_1)$	...	$\varphi(a_m; q_j)$; $\psi(a_m; q_j)$	...	$\varphi(a_m; q_n)$; $\psi(a_m; q_n)$

Задание автомата диаграммой Мура

Другой способ задания конечного автомата — графический. При этом способе состояния автомата изображают кружками, в которые вписывают символы состояний q_j $(j = 1, \ldots, n)$. Из каждого кружка проводится m стрелок (ориентированных ребер) взаимно-однозначно соответствующих символам входного алфавита $X\{a_1, \ldots, a_m\}$. Стрелке, соответствующей букве $a_i \in X$ и выходящей из кружка $q_j \in Q$, приписывается пара $(a_i, \psi(a_i; q_j))$, причем эта стрелка ведет в кружок, соответствующий $\varphi(a_i, q_j)$.

Полученный рисунок называется графом автомата или, диаграммой Мура. Для не очень сложных автоматов этот способ более нагляден, чем табличный.

Задание конечного автомата системой булевых функций

Третий способ задания конечного автомата $A = (X; Q; Y; \varphi; \psi)$, заданного таблицей или диаграммой Мура, состоит в определении системы булевых функций.

Изложим алгоритм этого способа задания.

(1) Находим числа k, r, s, удовлетворяющие условиям $2^{k-1} < m < 2^k$; $2^{r-1} < n \leqslant 2^r$; $2^{s-1} < p \leqslant 2^s$, где $m = |X|$; $n = |Q|$; $p = |Y|$.

Очевидно, что k, r, s соответственно равны числу разрядов в двоичном представлении чисел m, n, p. Например, если $m = 5$, $n = 17$, $p = 3$, то $k = 3$, $r = 5$, $s = 2$.

(2) Кодирование состояний входных и выходных символов исходного автомата.

Каждому $q_j \in Q$ взаимно-однозначно ставим в соответствие двоичную последовательность длины r — двоичный код $\alpha(q) = z_1 z_2 \ldots z_r$. Аналогично каждому $a_i \in X$ и каждому $b_k \in Y$ ставим взаимно однозначно в соответствие двоичные последовательности $\beta(\alpha) = x_1 x_2 \ldots x_k$; $\gamma(b) = y_1 y_2 \ldots y_s$.

Отметим, что кодирование состояний, входных и выходных символов можно провести многими способами. При этом некоторые последовательности (коды) могут оказаться неиспользованными.

(3) Составляем следующую таблицу:

Код входного символа				Код текущего состояния				Код следующего состояния				Код выходного символа			
x_1	x_2	...	x_k	z_1	z_2	...	z_r	φ_1	φ_2	...	φ_r	ψ_1	ψ_2	...	ψ_s
0	0	...	0	0	0	...	0								
β_1	β_2		β_k	α_1	α_2		α_r	α'_1	α'_2		α'_r	γ_1	γ_2		γ_s
1	1	...	1	1	1	...	1								

Эта таблица содержит $k + r + r + s$ столбцов и 2^{k+r} строк. В первых $k + r$ столбцах выписаны все наборы длины $k + r$. Каждый такой набор соответствует паре (β, α), где α — возможный код некоторого состояния, β — код входного символа.

(4) Заполнение последних столбцов в таблице (предыдущий шаг).

Для каждой пары (a_i, q_j), где $a_i \in X$; $q_j \in Q$, находим код $\beta(a)$ и $\alpha(q)$. По таблице автомата (или диаграмме Мура) определяем $\varphi(a; q) = q'$ и $\psi(a; q) = b$. Затем находим код $\alpha(q') = \alpha'_1\alpha'_2 \ldots \alpha'_r$ и код $\gamma(\beta) = \gamma_1\gamma_2 \ldots \gamma_s$. В строку таблицы, соответствующую набору

$$\beta_1, \beta_2, \ldots, \beta_k, \alpha_1, \alpha_2, \ldots, \alpha_r,$$

дописываем набор

$$\alpha'_1, \alpha'_2, \ldots, \alpha'_r, \gamma_1, \gamma_2, \ldots, \gamma_s.$$

(5) Определение системы булевых функций.

После выполнения предыдущего шага может оказаться, что не все строки в таблице заполнены. Это произойдет в том случае, если хотя бы одно из чисел m, n не является степенью 2. Таким образом, функции $\varphi_1, \varphi_2, \ldots, \varphi_r, \psi_1, \psi_2, \ldots, \psi_s$ окажутся не полностью определенными — на некоторых наборах их значения не определены. Тогда мы их доопределяем произвольным образом. Как правило, доопределение функций производят так, чтобы получившиеся полностью определенные функции удовлетворяли тем или иным условиям оптимальности, например представлялись минимальными ДНФ.

После выполнения этого шага исходный автомат будет задаваться системой полностью определенных булевых функций

$$\{\varphi_1, \varphi_2, \ldots, \varphi_r, \psi_1, \psi_2, \ldots, \psi_s\}.$$

39. Примеры конечных автоматов.

Пример 1. Элемент задержки (элемент памяти).

Элементы задержки представляют собой устройство, имеющее один вход и один выход. Причем значение выходного сигнала в момент времени t совпадает со значением сигнала в предыдущий момент. Схематично элемент задержки можно изобразить следующим образом (рис. 7.2).

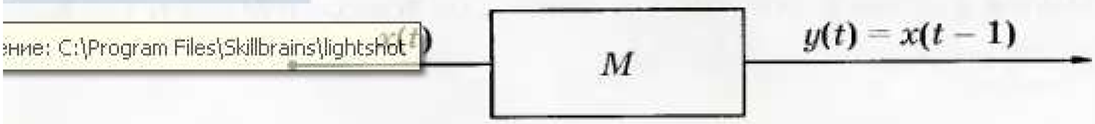


Рис. 7.2

Предположим, что входной и, следовательно, выходной алфавит есть $X = \{0, 1\}$; $Y = \{0, 1\}$. Тогда $Q = \{0, 1\}$. Под состоянием элемента задержки в момент времени t понимается содержание элемента памяти в данный момент. Таким образом $q(t) = X(t-1)$, а $Y(t) = q(t) = X(t-1)$.

Зададим элемент задержки таблицей, где $a_1 = 0$, $a_2 = 1$, $q_1 = 0$, $q_2 = 1$,

$$\begin{aligned}
&\varphi(a_1; q_1) = \varphi(0, 0) = 0; \quad &&\psi(a_1, q_1) = \psi(0, 0) = 0;\\
&\varphi(a_1; q_2) = \varphi(0, 1) = 0; \quad &&\psi(a_1, q_2) = \psi(0, 1) = 1;\\
&\varphi(a_2; q_1) = \varphi(1, 0) = 1; \quad &&\psi(a_2, q_1) = \psi(1, 0) = 0;\\
&\varphi(a_2; q_2) = \varphi(1, 1) = 1; \quad &&\psi(a_2, q_2) = \psi(1, 1) = 1,
\end{aligned}$$

a \ q	0	1
0	$\varphi = 0; \psi = 0$	$\varphi = 0; \psi = 1$
1	$\varphi = 1; \psi = 0$	$\varphi = 1; \psi = 1$

Диаграмма Мура изображена на рис. 7.3.

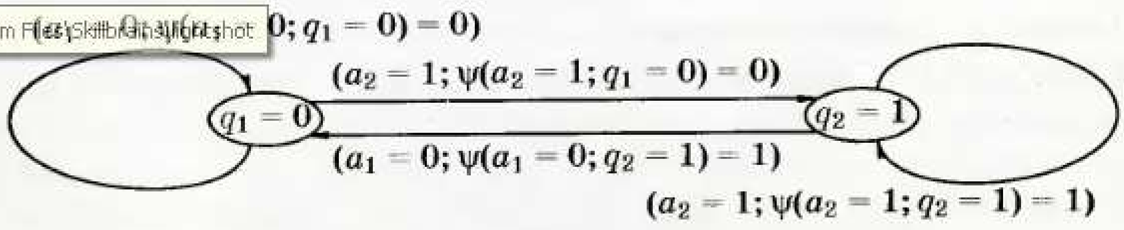


Рис. 7.3

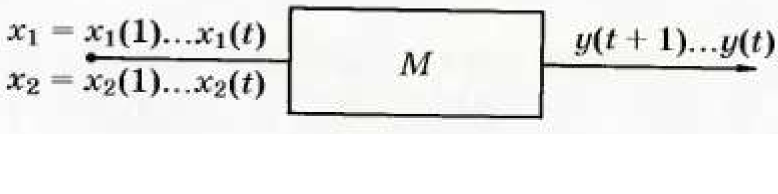
Для представления этого автомата системой булевых функций используем таблицу автомата и вышеизложенный алгоритм. При этом кодирование производить не нужно, так как входной и выходной алфавиты и состояния уже закодированы.

Обратим внимание на то, что $m = n = p = 2$. Тогда $k = r = s = 1$, и поэтому элемент задержки задается двумя функциями φ и ψ. Таблица истинности этих функций содержит $2^{k+r} = 2^2 = 4$ строки и $k + r + r + s = 4$ столбца:

x	z	φ	ψ
0	0	0	0
0	1	0	1
1	0	1	0
1	1	1	1

Пример 2. Двоичный сумматор последовательного действия.

Данный сумматор последовательного действия представляет собой устройство, осуществляющее сложение двух чисел в двоичной системе исчисления. На входы сумматора подаются числа x_1 и x_2, начиная с младших разрядов. На выходе формируется последовательность, соответствующая записи числа $x_1 + x_2$ в двоичной системе исчисления (рис. 7.4).



Входной и выходной алфавиты определены однозначно: $X = \{00; 01; 10; 11\}$; $Y = \{0, 1\}$. Множество состояний определяется значением переноса при сложении соответствующих разрядов чисел x_1 и x_2. Если при сложении некоторых разрядов образовался перенос, то будем считать, что сумматор перешел в состояние q_1. При отсутствии переноса будем считать, что сумматор находится в состоянии q_0.

Сумматор задается таблицей.

$a \backslash q$	q_0	q_1
00	$q_0; 0$	$q_0; 1$
01	$q_0; 1$	$q_1; 0$
10	$q_0; 1$	$q_1; 0$
11	$q_1; 0$	$q_1; 1$

Диаграмма Мура сумматора последовательного действия изображена на рис. 7.5.

Заметим, что входные и выходные символы уже закодированы. Состояния закодируем следующим образом: $\alpha(q_0) = 0$; $\alpha(q_1) = 1$. Поэтому

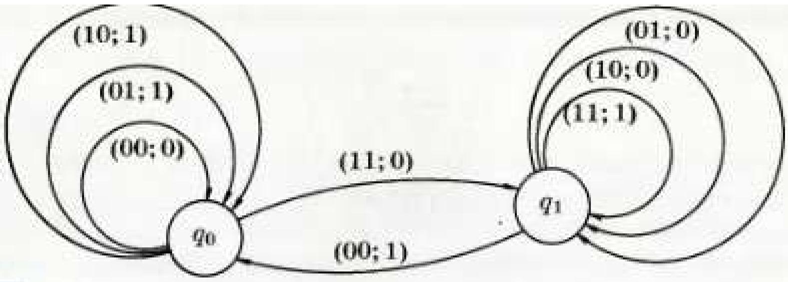


Рис. 7.5

сумматор последовательного действия задается двумя булевыми функциями, таблица истинности которых следующая:

x_1	x_2	z	φ	ψ
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

Пример 3. Схема сравнения на равенство.

Схема сравнения на равенство представляет собой устройство, сравнивающее два числа x_1 и x_2, заданные в двоичной системе исчисления. Это устройство работает следующим образом. На вход устройства последовательно, начиная со старших, подаются разряды чисел x_1 и x_2. Эти разряды сравниваются. При совпадении разрядов на выходе схемы формируется выходной сигнал 0, в противном случае на выходе появляется сигнал 1. Ясно, что появление 1 в выходной последовательности означает, что сравниваемые числа x_1 и x_2 различны. Если же выходная последовательность является нулевой и ее длина совпадает с числом разрядов сравниваемых чисел, то $x_1 = x_2$.

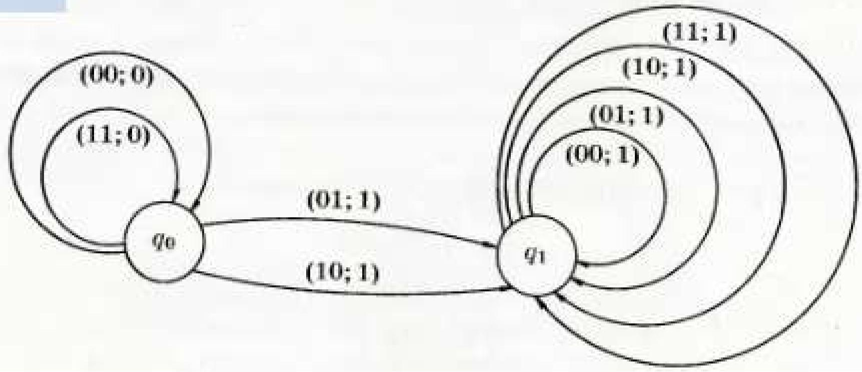
Для этого автомата $X = \{00; 01; 10; 11\}$; $Y = \{0, 1\}$.

Функционирование схемы определяется двумя состояниями. Состояние q_0 соответствует равенству сравниваемых в данный момент разрядов. При этом автомат остается в этом же состоянии. Если в следующий момент сравниваемые разряды будут различны, то автомат перейдет в новое состояние q_1 и в нем остается. Так как это означает, что числа различны.

Таким образом, схему сравнения можно задать таблицей:

$x \backslash q$	q_0	q_1
00	q_0; 0	q_1; 1
01	q_1; 1	q_1; 1
10	q_1; 1	q_1; 1
11	q_0; 0	q_1; 1

Диаграмма Мура схемы сравнения на равенство изображена на рис.



Кодирование состояний произведем следующим образом: $\alpha(q_0) = 0$; $\alpha(q_1) = 1$. Автомат будет задаваться двумя функциями.

x_1	x_2	z	φ	ψ
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	1	1
1	0	0	1	1
1	0	1	1	1
1	1	0	0	0
1	1	1	1	1

Пример 4. Схема сравнения на неравенство.

Схема сравнения на неравенство представляет собой устройство, позволяющее выяснить, равны ли сравниваемые x_1 и x_2, и если они не равны, выяснить, какое из них больше другого. Это устройство имеет два входа и два выхода. Выходные сигналы $y_1(t)$ и $y_2(t)$ определяются по следующим правилам:

$$y_1(t) = y_2(t) = 0, \text{ если } x_1(t) = x_2(t);$$

$$y_1(t) = 1; \quad y_2(t) = 0, \text{ если } x_1(t) > x_2(t); \quad \text{т. е. } x_1(t) = 1; \quad x_2(t) = 0;$$

$$y_1(t) = 0; \quad y_2(t) = 1, \text{ если } x_1(t) < x_2(t); \quad \text{т. е. } x_1(t) = 0; \quad x_2(t) = 1.$$

Таким образом, при подаче на вход схемы сравнения на неравенство чисел $x_1 = x_1(1)\ldots x_1(t)$ и $x_2 = x_2(1)\ldots x_2(t)$ последовательно сравниваются разряды этих чисел, начиная со старших. Выходные сигналы формулируются согласно вышеуказанным правилам. При этом, если выходная последовательность состоит из нулевых пар, то $x_1 = x_2$. Если первая, отличная от нулевой, пара имеет вид $\begin{pmatrix}1\\0\end{pmatrix}$, ($\begin{pmatrix}0\\1\end{pmatrix}$) то $x_1 > x_2$ ($x_1 < x_2$).

Из описания схемы следует, что

$$X = \{00; 01; 10; 11\}; \quad Y = \{00; 01; 10\}.$$

Состояние схемы определяется следующим образом. Предположим, что в начальный момент времени $t = 1$ автомат находится в состоянии q_1. Если сравниваемые разряды чисел x_1 и x_2 совпадают, то автомат остается в этом состоянии. Заметим, что на выходе при этом появится сигнал 00. Если же разряд числа x_1 будет меньше (больше) соответствующего разряда числа x_2, то автомат перейдет в состояние q_2 (q_3). При этом на выходе появится сигнал 01 (10). В дальнейшем при подаче оставшихся разрядов чисел x_1 и x_2 на входы автомата автомат будет оставаться в состоянии q_2 (q_3) и вырабатывать выходной символ 10 (01). Из вышеизложенного следует, что схему сравнения на неравенство можно задать таблицей:

$x \backslash q$	q_1	q_2	q_3
00	q_1; 00	q_2; 01	q_3; 10
01	q_2; 01	q_2; 01	q_3; 10
10	q_3; 10	q_2; 01	q_3; 10
11	q_1; 00	q_2; 01	q_3; 10

Соответствующая диаграмма Мура изображена на рис. 7.7.

Входной и выходной алфавиты здесь уже закодированы. Состояния q_1, q_2 и q_3 закодируем: $\alpha_1(q_1) = 00$; $\alpha(q_2) = 01$; $\alpha(q_3) = 10$.

Следовательно, данную схему можно задать системой, состоящей из четырех булевых функций, которые зависят от четырех переменных. Эти

функции частично определены и задаются таблицей истинности

В таблице символами * отмечены наборы переменных x_1, x_2, z_1, z_2, на которых функции φ_1, φ_2, ψ_1, ψ_2 не определены. Положим значения функций φ_1, φ_2, ψ_1, ψ_2, на этих наборах равными 1.

	x_1	x_2	z_1	z_2	φ_1	φ_2	ψ_1	ψ_2
	0	0	0	0	0	0	0	0
	0	0	0	1	0	1	0	1
	0	0	1	0	1	0	1	0
*	0	0	1	1	1	1	1	1
	0	1	0	0	0	1	0	1
	0	1	0	1	0	1	0	1
	0	1	1	0	0	1	0	1
*	0	1	1	1	1	1	1	1
	1	0	0	0	1	0	1	0

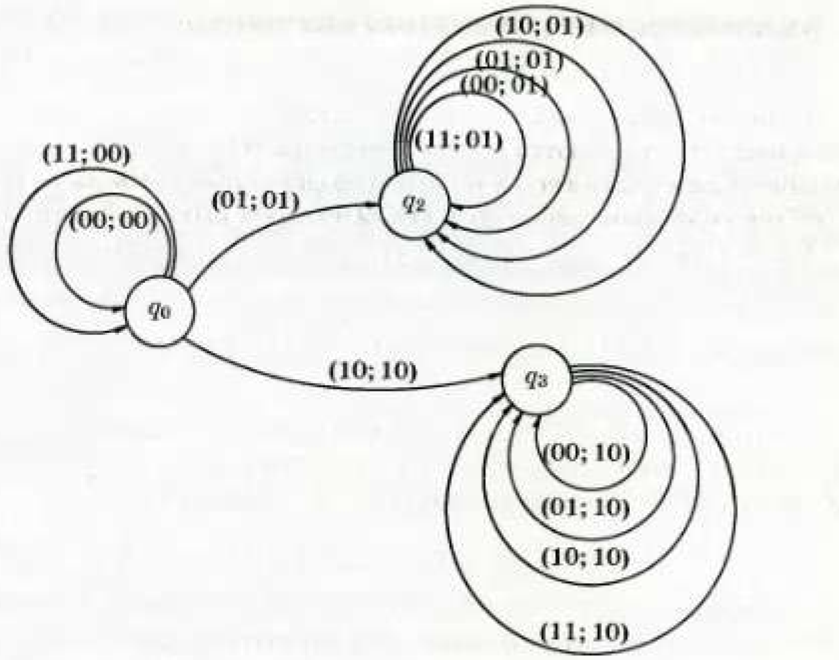


рис. 7.7

40. Канонические уравнения автомата.

Если в момент времени $t = 1, 2, \ldots$ на вход автомата $A = (X; Q; Y; \varphi; \psi)$ последовательно подаются входные символы $x(t) \in X$ и при этом автомат находится в состоянии $q(t) \in Q$, то под воздействием символа $x(t)$ автомат перейдет в новое состояние $q(t+1) \in Q$ и выдаст выходной сигнал $y(t)$.

Величины $x(t)$, $y(t)$, $q(t)$, $q(t+1)$ связаны между собой следующими уравнениями:

$$\begin{cases} q(t+1) = \varphi\,(x(t); q(t)), \\ y(t) = \psi\,(x(t); q(t)), \end{cases} \qquad t = 1, 2, \ldots, n, \ldots$$

Эти уравнения называются каноническими уравнениями автомата A. При задании автомата системой булевых функций эти уравнения записываются в координатной форме:

$$\begin{gathered} z_1(t+1) = \varphi_1\,(x_1(t), \ldots, x_k(t), z_1(t), \ldots, z_r(t)) \\ \ldots \\ z_r(t+1) = \varphi_r\,(x_1(t), \ldots, x_k(t), z_1(t), \ldots, z_r(t)) \\ y_1(t) = \psi_1\,(x_1(t), \ldots, x_k(t), z_1(t), \ldots, z_r(t)) \\ \ldots \\ y_s(t) = \psi_s\,(x_1(t), \ldots, x_k(t), z_1(t), \ldots, z_r(t)) \\ t = 1, 2, \ldots \end{gathered}$$

Для построения канонических уравнений автомата A необходимо для данной булевой функции найти минимальную ДНФ (дизъюнктивную нормальную форму), которая, вообще говоря, определяется неоднозначно. Аналитический алгоритм построения этой ДНФ следующий:

1. Для данной функции $f(x_1, \ldots, x_n)$ строим совершенную конъюнктивную нормальную форму (СКНФ).

2. В построенной СКНФ раскрываем скобки, используя правило:

$$(A \vee B) \wedge (C \vee D) = A \wedge C \vee B \wedge C \vee A \wedge D \vee B \wedge D.$$

3. Полученное выражение упрощаем, применяя тождества вида:

$$K_1 \wedge K_2 \vee K_1 = K_1; \quad K \vee K = K; \quad K \vee 0 = K; \quad K \wedge \overline{K} = 0; \quad K \vee 1 = 1;$$
$$K \wedge 1 = K; \quad K \wedge K = K; \quad K \vee \overline{K} = 1.$$

В результате получим сокращенную ДНФ, являющуюся дизъюнкцией всех простых импликат данной функции $f(x_1, \ldots, x_n)$.

Для рассмотренных выше примеров автоматов канонические уравнения задаются следующими формулами:

пример 1: $\begin{cases} z(t+1) = x(t), \\ y(t) = z(t), \end{cases} \quad t = 1, 2, \ldots$

пример 2: $\begin{cases} z(t+1) = x_1(t) \wedge x_2(t) \vee x_1(t) \wedge z(t) \vee x_2(t) \wedge z(t), \\ y(t) = x_1(t) \oplus x_2(t) \oplus z(t), \end{cases} \quad t = 1, 2, \ldots$

пример 3: $\begin{cases} z(t+1) = \overline{x_1}(t) \wedge x_2(t) \vee x_1(t) \wedge \overline{x_2}(t) \vee z(t), \\ y(t) = \overline{x_1}(t) \wedge x_2(t) \vee x_1(t) \wedge \overline{x_2}(t) \vee z(t), \end{cases} \quad t = 1, 2, \ldots$

пример 4: $\begin{cases} z_1(t+1) = z_1(t) \vee x_1(t) \wedge \overline{x_2}(t) \wedge \overline{z_2}(t), \\ z_1(t+1) = z_2(t) \vee \overline{x_1}(t) \wedge x_2(t) \wedge \overline{z_1}(t), \\ y_1(t) = z_1(t) \vee x_1(t) \wedge \overline{x_2}(t) \wedge \overline{z_2}(t), \\ y_2(t) = z_2(t) \vee \overline{x_1}(t) \wedge x_2(t) \wedge \overline{z_1}(t), \end{cases} \quad t = 1, 2, \ldots$

В качестве иллюстрации изложенного выше алгоритма рассмотрим пример 3.

Таблица истинности системы булевых функций следующая:

x_1	x_2	z	φ	ψ
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	1	1
1	0	0	1	1
1	0	1	1	1
1	1	0	0	0
1	1	1	1	1

1. Строим СКНФ функции $\varphi(x_1, x_2, z)$. Так как эта функция задана набором своих значений $\overline{\varphi} = 01111101$, то ее СКНФ будет иметь вид $(x_1 \vee x_2 \vee z) \wedge (\overline{x_1} \vee \overline{x_2} \vee z)$.

2. Раскрываем скобки

$$(x_1 \vee x_2 \vee z) \wedge \overline{x_1} \vee (x_1 \vee x_2 \vee z) \wedge \overline{x_2} \vee (x_1 \vee x_2 \vee z) \wedge z =$$
$$= x_1 \wedge \overline{x_1} \vee x_2 \wedge \overline{x_1} \vee z \wedge \overline{x_1} \vee x_1 \wedge \overline{x_2} \vee x_2 \wedge \overline{x_2} \vee z \wedge \overline{x_2} \vee x_1 \wedge z \vee x_2 \wedge z \vee z \wedge z.$$

Упрощаем последнее выражение:

$$0 \vee \overline{x_1} \wedge x_2 \wedge \overline{x_1} \wedge z \vee x_1 \wedge \overline{x_2} \vee 0 \vee \overline{x_2} \wedge z \vee x_1 \wedge z \vee x_2 \wedge z \vee z =$$
$$= \overline{x_1} \wedge x_2 \vee x_1 \wedge \overline{x_2} \vee z(\overline{x_2} \vee x_2) \vee z(\overline{x_1} \vee x_1) \vee z = \overline{x_1} \wedge x_2 \vee x_1 \wedge \overline{x_2} \vee z.$$

Таким образом, получим $z(t+1) = \overline{x_1}(t) \wedge x_2(t) \vee x_1(t) \wedge \overline{x_2}(t) \vee z(t)$.

Аналогично строится функция $y(t)$. При этом из таблицы истинности выписываем набор значений функции $\psi(x_1, x_2, z)$ $\overline{\psi} = 01111101$, который совпадает с набором значений функции $\varphi(x_1, x_2, z)$.

6.2. Эталоны ответов на практические задания:

1. Докажите тождественную истинность формулы $\overline{X}$ **→(X→У).**

Решение. Составим таблицу истинности:

X	Y	$\overline{X}$	X→У	$\overline{X}$ →(X→У).
0	0	1	1	1
0	1	1	1	1
1	0	0	0	1
1	1	0	1	1

Ответ: последний столбец состоит из 1, следовательно, доказана тождественная истинность формулы.

2. С помощью таблиц истинности проверить, являются ли эквивалентными высказывания: **f_1= X˄ (Y→Z) и f_2= ($\overline{X}$˄Y) ˅ (X˄Z).**

Решение.

X	**Y**	**Z**	**$\overline{X}$**	**Y→Z**	**f_1**	**$\overline{X}$˄Y**	**X˄Z**	**f_2**
0	0	0	1	1	0	0	0	0
0	0	1	1	1	0	0	0	0
0	1	0	1	0	0	1	0	1
0	1	1	1	1	0	1	0	1

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

Ответ: так как значения для высказываний f_1 и f_2 в таблице истинности не совпадали, то они не эквивалентны.

3. Определите для каждого из следующих высказываний, будет ли оно логически истинным, противоречивым: ни тем, ни другим.

а) $X\leftrightarrow X$, б) $X\leftrightarrow\overline{X}$, в) $(X\vee У)\leftrightarrow(X\wedge У)$, г) $(X\to\overline{У})\to(У\to\overline{X})$,

д) $(X\to Y)\wedge(Y\to Z)\wedge\overline{(X\to Z)}$, е) $(X\to У)\to X$, ж) $((X\to У)\to X$.

Решение.

X	Y	**$X\leftrightarrow X$**	$\overline{X}$	**$X\leftrightarrow\overline{X}$**	$X\vee У$	$X\wedge У$	**$(X\vee У)\leftrightarrow(X\wedge У)$**
0	0	**1**	1	**0**	0	0	**1**
0	1	**1**	1	**0**	1	0	**0**
1	0	**1**	0	**0**	1	0	**0**
1	1	**1**	0	**0**	1	1	**1**
		а)		**б)**			**в)**

Вывод: а) – логически истинное, б) – противоречивое, в) – ни то, ни другое.

X	У	$\overline{У}$	$X\to\overline{У}$	$У\to\overline{X}$	**$(X\to\overline{У})\to(У\to\overline{X})$**
0	0	1	1	1	**1**
0	1	0	1	1	**1**
1	0	1	1	1	**1**
1	1	0	0	0	**1**
					г)

г) – логически истинное

X	У	Z	$X\leftrightarrow У$	$У\leftrightarrow Z$	$(X\to Y)\wedge(Y\to Z)$	$X\to Z$	$\overline{X\to Z}$	**$(X\to Y)\wedge(Y\to Z)\wedge\overline{(X\to Z)}$**
0	0	0	1	1	1	1	0	**0**

0	0	1	1	1	1	1	0	**0**
0	1	0	1	0	0	1	0	**0**
0	1	1	1	1	1	1	0	**0**
1	0	0	0	1	0	0	1	**0**
1	0	1	0	1	0	1	0	**0**
1	1	0	1	0	0	0	1	**0**
1	1	1	1	1	1	1	0	**0**
								д)

д) – противоречиво

Х	У	Х→У	**(Х→У) →Х**	**((Х→У) →Х) →Х**
0	0	1	**0**	**1**
0	1	1	**0**	**1**
1	0	0	**1**	**1**
1	1	1	**1**	**1**
			е)	**ж)**

е) – ни то, ни другое; ж) – логически истинно

Ответ: а) – логически истинное, б) – противоречивое, в) – ни то, ни другое, г) – логически истинное, д) – противоречиво, е) – ни то, ни другое; ж) – логически истинно.

4. Доказать закон отрицания конъюнкции **(**$\overline{A \wedge B} \leftrightarrow \overline{A} \vee \overline{B}$**)**

Доказательство:

Найдем значения для $\overline{A \wedge B}$ и $\overline{A} \vee \overline{B}$ и сравним их.

A	**B**	$A \wedge B$	$\overline{A \wedge B}$	$\overline{A}$	$\overline{B}$	$\overline{A} \vee \overline{B}$
И	И	И	**Л**	Л	Л	**Л**
И	Л	Л	**И**	Л	И	**И**
Л	И	Л	**И**	И	Л	**И**
Л	Л	Л	**И**	И	И	**И**

Ответ: значения выражений совпадают, следовательно, закон доказан.

5. Найти значение $\overline{A \wedge B} \leftrightarrow \overline{A} \vee \overline{B}$ и убедиться, что при всех значениях А и В - это истинное значение.

Решение.

A	**B**	$A \wedge B$	$\overline{A \wedge B}$	$\overline{A}$	$\overline{B}$	$\overline{A} \vee \overline{B}$	$\overline{A \wedge B} \leftrightarrow \overline{A} \vee$
И	И	И	Л	Л	Л	Л	**И**
И	Л	Л	И	Л	И	И	**И**
Л	И	Л	И	И	Л	И	**И**
Л	Л	Л	И	И	И	И	**И**

Ответ: последний столбец состоит из «И», следовательно, доказана тождественная истинность формулы.

6. С помощью основных равносильностей доказать закон обобщенного склеивания $xy \vee \bar{x}z = xy \vee \bar{x}z \vee yz$.

Доказательство. Применяя закон склеивания (в обратном порядке, то есть $yz = xyz \vee \bar{x}yz$) и дистрибутивность (то есть вынесем за скобки xy и $\bar{x}z$), получим $xy \vee \bar{x}z \vee yz = xy \vee \bar{x}z \vee xyz \vee \bar{x}yz = xy(1 \vee z) \vee \bar{x}z(1 \vee y) = xy \vee \bar{x}z$.

7. Составьте таблицу истинности булевой функции трех переменных $f(x_1, x_2, x_3) =$

$x_1 \oplus x_2 \rightarrow \overline{x3} \vee x_1 \,|\, (\overline{x2} \wedge \overline{x1})$ и найдите ее двоичный набор.

Решение. Для вычисления значений функции следует определить порядок выполнения операций. Это можно сделать многими способами. Пусть, например, порядок выполнения операций будет следующим:

$f_1 = x_1 \oplus x_2$; $f_2 = \overline{x2} \wedge \overline{x1}$; $f_3 = x_1 \,|\, f_2$; $f_4 = \overline{x3} \vee f_3$; $f_5 = f_1 \rightarrow f_4$.

Последовательно составляются таблицы истинности всех указанных функций.

x_1	x_2	x_3	$\overline{x1}$	$\overline{x2}$	$\overline{x3}$	f_1	f_2	f_3	f_4	f_5
0	0	0	1	1	1	0	1	1	1	1
0	0	1	1	1	0	0	1	1	1	1
0	1	0	1	0	1	1	0	1	1	1
0	1	1	1	0	0	1	0	1	1	1
1	0	0	0	1	1	1	0	1	1	1
1	0	1	0	1	0	1	0	1	1	1
1	1	0	0	0	1	0	0	1	1	1
1	1	1	0	0	0	0	0	1	1	1

Лексикографическое упорядочение наборов в таблице истинности булевой функции позволяет задать функцию двоичным набором длины 2^n, который будет обозначаться буквой F. Двоичный набор данной функции F = 11111111. Отметим, что двоичный набор определяет булеву функцию в том и только в том случае, когда его длина есть степень двойки, а соответствующий показатель степени определяет число переменных данной функции.

8. Докажите эквивалентность функции: $\mathbf{f_{(x,\,y,\,z)} = x \wedge (x \vee z) \wedge (y \vee z)}$ **и** $\mathbf{f_{(x,\,y,\,z)} = (x \wedge y) \vee (x \wedge z)}$**.**

Решение. Для доказательства необходимо поострить таблицы истинности этих функций, и если их двоичные наборы совпадут, то эквивалентность будет доказана.

x	y	z	$x \vee z$	$y \vee z$	$x \wedge (x \vee z)$	$x \wedge (x \vee z) \wedge (y \vee z)$
0	0	0	0	0	0	0
0	0	1	1	1	0	0
0	1	0	0	1	0	0
0	1	1	1	1	0	0
1	0	0	1	0	0	0
1	0	1	1	1	1	1
1	1	0	1	1	1	1
1	1	1	1	1	1	1

x	y	z	$x \wedge y$	$z \wedge x$	$(x \wedge y) \vee (x \wedge z)$
0	0	0	0	0	0
0	0	1	0	0	0
0	1	0	0	0	0
0	1	1	0	0	0
1	0	0	0	0	0
1	0	1	0	1	1
1	1	0	1	0	1
1	1	1	1	1	1

Получаем F_1 = 00000111 и F_2 = 00000111. Значит, функции эквивалентны.

Ответ: функции эквивалентны.

9. Найдите СДНФ и СКНФ функции **$f_{(x, y, z)}$,** заданной следующей таблицей истинности:

x_1	x_2	x_3	$f(x_1, x_2, x_3)$
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

Решение. По теореме о функциональной полноте **СДНФ** имеет вид:

СКНФ имеет вид:

$(x_1 \vee x_2 \vee \overline{x3}) \wedge (x_1 \vee \overline{x2} \vee x_3) \wedge (\overline{x1} \vee x_2 \vee x_3) \wedge (\overline{x1} \vee \overline{x2} \vee x_3)$.

Описанный способ нахождения СДНФ и СКНФ по таблице истинности бывает часто более трудоемким. Для нахождения СДНФ данную формулу приводим сначала к ДНФ, а затем преобразовываем ее конъюнкции с помощью следующих действий:

А) Если в конъюнкцию входит некоторая переменная со своим отрицанием, то мы удаляем эту конъюнкцию из ДНФ;

Б) Если конъюнкцию одна и та же переменная входит несколько раз, то все они удаляются, кроме одной;

В) Если в конъюнкцию не входят некоторые переменные, то для каждой из них к конъюнкции добавляется соответствующая формула вида $(x \vee \overline{x})$;

Г) Если в полученной ДНФ имеется несколько одинаковых конъюнкций, т оставляем только одну из них. В результате получается СДНФ.

10. Опрос 100 студентов выявил следующие данные о числе студентов, изучающих различные иностранные языки: английский – 28; немецкий - 30; французский – 42, английский и немецкий – 8; английский и французский – 10; немецкий и французский – 5; все три языка – 3.

1) Сколько студентов не изучают ни одного иностранного языка?

2) Сколько студентов изучают один французский язык?

3) Сколько студентов изучают немецкий язык в том и только в том случае, если они изучают французский язык?

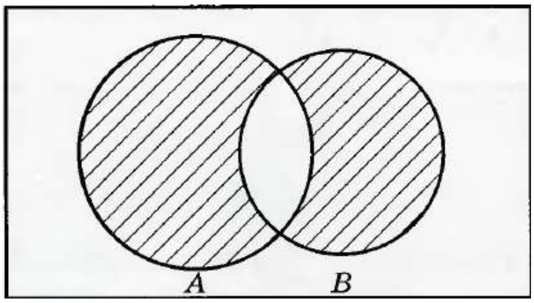
Решение. Нарисовать диаграмму Эйлера-Венна в виде трех кругов, обозначающих множество студентов, изучающих соответственно французский, немецкий и английский языки. В каждую из 8-ми областей вписать данные, используя приведенные цифры. Начинать с конца списка и двигаться к началу.

Ответ: 1) 20; 2) 30; 3) 38.

11. Изобразите с помощью диаграмм Эйлера-Венна множества:

1) $A \subset B$ и $B \subset C$; 2) $A \subset B$; $B \subset C$ и $A \setminus B = \emptyset$; 3) $A \subset B$; $B \subset C$ и $C = A \cup B$;
4) $A \subset B$; $B \subset C$ и $A \cap B \neq \emptyset$; 5) $(A \setminus B) \cap (B \setminus A)$.

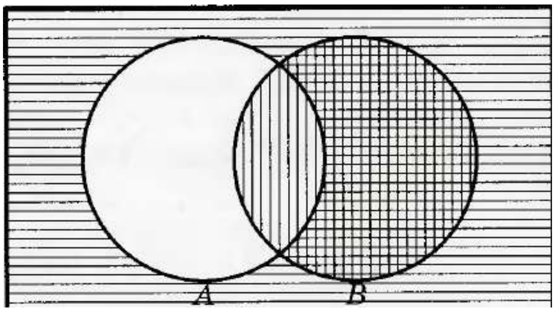
Решение. Изобразим с помощью диаграмм Эйлера-Венна множество
$(A \setminus B) \cap (B \setminus A)$



Это множество является объединением двух разностей, называется симметрической разностью и обозначается $A \oplus B$, т.е. (A \ B) ∩(B \ A) = $A \oplus B$.

12. Воспользовавшись диаграммой Эйлера-Венна, определите, какие из следующих высказываний истинны: **а)** $\mathbf{X \vee \overline{X}}$**; б)** $\mathbf{X \wedge \overline{X}}$**; в)** $\mathbf{X} \vee (\overline{X} \wedge \boldsymbol{Y})$; **г) X→(Y→X); д)**$\mathbf{X} \wedge \overline{(\boldsymbol{Y} \to \boldsymbol{X})}$**.**

Решение. Множество истинности высказывания X→Y совпадает с заштрихованной областью на диаграмме:



Т.к.эта заштрихованная область включает в себя множество B, то мы видим, что из Y следует X→Y.

13. Пусть A = {1, 2}. Выписать все элементы декартова произведения A×A.

Решение. A×A = {‹1, 1› ; ‹1, 2› ; ‹2, 1› ; ‹2, 2›}.

14. Рассмотрим два множества A = {a, b, c, d, e, f, g, h} и B = {1, 2, 3, 4, 5, 6, 7, 8}.

Составьте множество пар ‹x, y› ∈A×B. Что это множество представляет?

Ответ: множество клеток шахматной доски.

15. Проверьте на линейность функцию $f(x_1, x_2, x_3)$, если ее двоичный набор F= 11100001.

Решение. Применяем к функции $f(x_1, x_2, x_3)$ алгоритм проверки на линейность.

Вычисляем коэффициенты a_0, a_1, a_2, a_3 многочлена Жегалкина для данной функции:

$a_0 = f(0, 0, 0,) = 1$; $a_1 = f(0, 0, 0)\oplus f(1, 0, 0) = 1\oplus 0 = 1$; $a_2 = f(0, 0, 0)\oplus f(0, 1, 0) = 1\oplus 1 = 0$; $a_3 = f(0, 0, 0)\oplus f(0, 0, 1) = 1\oplus 1 = 0$.

Вычисляем многочлен: $f(x_1, x_2, x_3) = a_0\oplus a_1x_1\oplus a_2x_2\oplus a_3x_3 = x_1\oplus 1$.

Очевидно, что двоичный набор F = 11110000 многочлена $f(x_1, x_2, x_3) = x_1\oplus 1$ не совпадает с двоичным набором булевой функции, следовательно, функция $f(x_1, x_2, x_3)$ не линейна.

16. Найдите правую и левую области отношения R = {‹1, 5› ; ‹1, 6› ; ‹1, 7›}.

Решение. $D_{пр.}$ = {5, 6, 7}; $D_{лев}$ = {1}.

Ответ: $D_{пр.}$ = {5, 6, 7}; $D_{лев}$ = {1}.

17. Если A = {2, 3,4, 5, 6, 7, 8}, запишите бинарное отношение R = {‹x, y› : x, y $\in$ A,

x делит y, и x ≤ 3}.

Ответ: R = {‹2, 2› ; ‹2, 4› ; ‹2, 6› ; ‹2, 8› ; ‹3, 3› ; ‹3, 6›}.

18. Являются ли следующие отношения функциями:

2) {‹1, 2› ; ‹2, 3› ; ‹3, 2›}; 2) {‹1, 2› ; ‹1, 3› ; ‹2, 3›};
3) $\{x, x^2 - 2x - 3 : x \in R\}$?

Решение. Отношение 1) – функция; отношение 2)- не является функцией;

отношение 3) -функция, которая обычно обозначается $y = x^2 - 2x - 3$.

19. Футбольный мяч сшит из 32 лоскутов: белых шестиугольников и чёрных пятиугольников. Каждый чёрный лоскут граничит только с белыми, а каждый белый – с тремя чёрными и тремя белыми. Сколько лоскутов белого цвета?
Решение. Пусть на мяче число белых лоскутов x штук, а чёрных будет (32 – x) штук. Тогда, границ белых лоскутов с чёрными: 5(32-x) =3x, значит x= 20 и белых лоскутов 20 штук.
Ответ: 20 лоскутов белого цвета.

20. Из 4 первокурсников, 5 второкурсников и 6 третьекурсников надо выбрать 3 студента на конференцию. Сколькими способами можно осуществить этот выбор, если среди выбранных должны быть студенты разных курсов?

Решение .$A_4^1\times A_5^1\times A_6^1$= 4*5*6 = 120

Ответ: 120 способов.

21. Сколько можно составить четырехзначных чисел так, чтобы любые две соседние цифры были различны?

Решение. Размещения с возвращением: 9^4=6561

Ответ: 6561 чисел.

22. Сколькими способами можно рассадить 5 человек за круглым столом (рассматривается только расположение сидящих относительно друг друга)?

Решение. P_4= 4! = 24

Ответ: 24 способа.

23. Сколькими способами можно распределить 15 выпускников по 3 районам, если в одном из них имеется 8, в другом - 5 и в третьем – 2 вакантных места?

Решение. $C_{15}^{8*} C_7^{5\,*}\ C_2^{2\,=}$ 135135.

Ответ: 135135 способов.

24. Известно, что 7 студентов сдали экзамен по дискретной математике на хорошо и отлично. Сколькими способами могли быть поставлены им оценки?

Решение. $\bar{A}_2^7$= 2^7=128 **Ответ: 128 способов.**

25. Группа студентов изучает 10 различных дисциплин. Сколькими способами можно составить расписание занятий в понедельник, если в этот день должно быть 4 разных занятия?

Решение. $\mathbf{A_{10}^4}$=10*9*8*7 = 5040 **Ответ: 5040 способов.**

26. Из 60 вопросов, входящих в экзаменационные билеты, студент знает 50. Найти вероятность того, что среди трех наугад выбранных вопросов студент знает:

а) все вопросы, б) два вопроса.

Решение. а) р = $\frac{C_{50}^3}{C_{60}^3}$ = 0,573; б) р = $\frac{C_{50}^2 * C_{10}^1}{C_{60}^3}$ = 0,36.

Ответ: а) 0,573; б) 0,36.

27. Во взводе три сержанта и 30 солдат. Сколькими способами можно выделить одного сержанта и трех солдат для патрулирования?

Решение. $C_3^1 * C_{30}^3$= 3!/1!2!* 30!/3!27! = 30*29*14 = 12180.

Ответ: 12180 способов.

28. В барабане револьвера 7 гнезд, из них в 5 заложены патроны. Барабан приводится во вращение, потом нажимается спусковой курок. Какова вероятность того, что, повторив такой опыт 2 раза подряд: а) револьвер оба раза не выстрелит, б) оба раза револьвер выстрелит.

Решение: а) P = $\frac{2*2}{7*7}$ = 4/49; б) P = $\frac{5*4}{49}$ = 20/49. **Ответ: 4/49; 20/49.**

29. Решить уравнение: $5C_x^3 = C_{x+2}^4$ **Ответ:** x = 14; x = 3.

30. Решить уравнение: $C_{x-3}^2 = 21$ **Ответ:** x = 10; x= - 8.

31. Решить уравнение: $A_x^3 = \frac{1}{20} A_x^4$ **Ответ**: x = 19/3.

32. Решить уравнение: $C_x^3 + C_x^2$ = 15(x-1) **Ответ:** x = -10; x = 9.

33. Решить уравнение: $C_x^3 = \frac{5x\,(x-3)}{4}$ **Ответ:** x= 1; x = 9,5.

34. Записать предложение: «прямая a параллельна прямой b» с помощью предиката.
Решение. Предметная область – множество прямых.
Введем предикат P (x), x – прямая. Предикат параллельности x||y .
Тогда предложение можно записать в виде: P (a) $\wedge$P(b) $\wedge$(a||b) .

35. Записать с помощью предиката: «Аксиома: через две различные точки проходит единственная прямая» (Если две точки принадлежат двум прямым, то эти прямые совпадают).

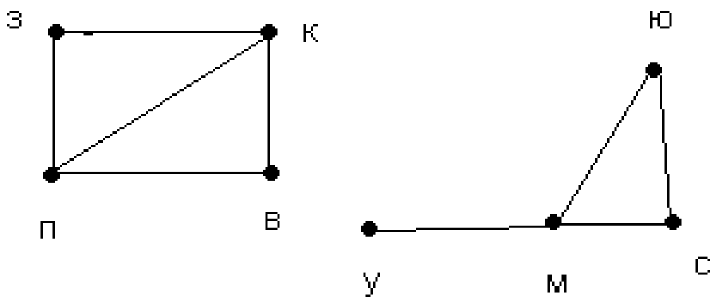
Решение. Введем предикаты**:** T (x), x – точка; P (x), x – прямая; J(x,y) - x $\in$ y.

Тогда можно записать:

T (A) $\wedge$T (B) $\wedge(A \neq B)$ $\wedge$P (a) $\wedge$P(b) $\wedge$J(A,a) $\wedge$J(B,a) $\wedge$J(A,b) $\wedge$J(B,b) $\rightarrow$(a=b).

36. Между планетами введено космическое сообщение по следующим маршрутам: З-К, П-В, З-П, П-К, К-В, У-М, М-С, С-Ю, Ю-М, М-У. Можно ли добраться с З до М?

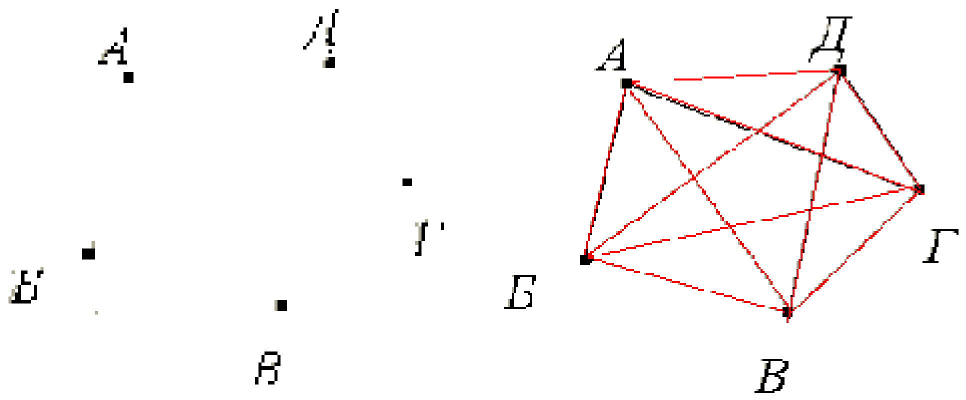
Решение: Составим схему-граф маршрутов:



Ответ: от З до М добраться нельзя.

37. Аркадий, Борис. Владимир, Григорий и Дмитрий при встрече обменялись рукопожатиями (каждый пожал руку каждому по одному разу). Сколько всего рукопожатий было сделано?

Решение. Пусть каждому из пяти молодых людей соответствует определенная точка на плоскости, названная первой буквой его имени, а производимому рукопожатию — отрезок или часть кривой, соединяющая конкретные точки — имена.



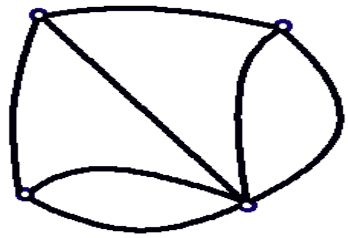
Если подсчитать число ребер графа, изображенного на рисунке справа, то это число и будет равно количеству совершенных рукопожатий между пятью молодыми людьми. Их 10. **Ответ: 10.**

38. К XVIII веку через реку, на которой стоял город Кенигсберг (ныне Калининград), было построено 7 мостов, которые связывали с берегами и друг с другом два острова, расположенные в пределах города.

Задача заключается в следующем: нужно пройти (если это возможно) по всем семи мостам так, чтобы на каждом из них побывать лишь по одному разу и вернуться к тому месту, откуда начал маршрут.

Решение:

Решить эту задачу удалось в 1736 г. Леонарду Эйлеру . В ходе решения задачи (после интерпретации условия задачи в виде графа, где вершины - острова и берега, а ребра - мосты, представленного на рисунке.)

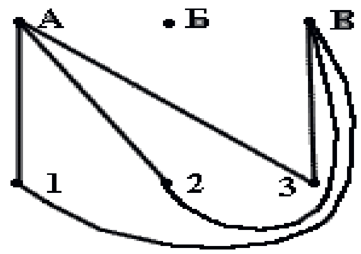


Прохождение по всем мостам при условии, что нужно на каждом побывать один раз и вернуться в точку начала путешествия, на языке теории графов выглядит как задача изображения «одним росчерком» графа, представленного на рисунке. Но, поскольку граф на этом рисунке имеет четыре нечетные вершины, то, согласно закономерности 7 такой граф начертить «одним росчерком» невозможно. Значит, и пройти по кенигсбергским мостам, соблюдая заданные условия, **нельзя**.

Ответ: нельзя.

39. В трех различных домах живут три поссорившиеся между собой соседа. Недалеко от их домов имеются три колодца. Можно ли от каждого дома проложить к каждому из колодцев тропинку так, чтобы никакие две из них не пересекались?

Решение. Построим граф, вершины которого А, Б, В, 1, 2, 3 соответствуют домам и колодцам условия задачи, и попробуем доказать, что девятую тропинку — ребро графа, не пересекающее остальные ребра, провести нельзя.

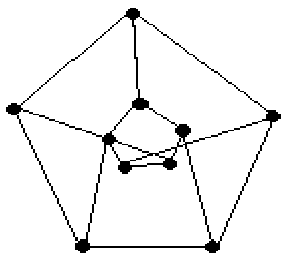


Проведенные в графе на рисунке ребра А1, А2, А3 и В1,В2, В3 (соответствующие тропинкам от домов А и В ко всем колодцам). Построенный граф разбил плоскость на три области: X, У, Z. Вершина Б, в зависимости от ее расположения на плоскости, попадает в одну из этих трех областей. Если вы рассмотрите каждый из трех случаев «попадания» вершины Б в одну из областей X, Y или Z, то убедитесь, что всякий раз одна из вершин графа 1, 2 или 3 (один из колодцев) будет «недоступной» для вершины Б (т. е. нельзя будет провести одно из ребер Б1, Б2 или Б3. которое

не пересекло бы уже имеющихся в графе ребер).
Таким образом, ответ на вопрос задачи будет таким: «Нельзя!»
Ответ: нельзя.

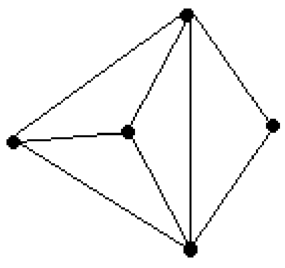
40. В городе Н от каждой площади отходит ровно пять улиц, соединяющих площади. Докажите, что число площадей чётно, а число улиц кратно пяти.
Решение. Из теоремы о том, что число нечётных вершин любого графа чётно, следует, что число площадей (вершин графа) 2n, а число улиц (рёбер графа) будет (2n·5):2, а значит, число площадей чётно, а число улиц кратно 5.

41. В государстве система авиалиний устроена таких образом, что любой город соединён авиалиниями не более чем с тремя другими и из любого города в любой другой можно проехать, сделав не более одной пересадки. Какое максимальное число городов может быть в этом государстве?
Решение. Пусть существует некоторый город А. Из него можно добраться не более, чем до трёх городов, а из каждого из них ещё не более чем до двух (не считая А). Тогда всего городов не более 1+3+6=10. Значит всего городов не более 10. Пример на рисунке (его ещё называют графом Петерсона) показывает существование авиалиний.

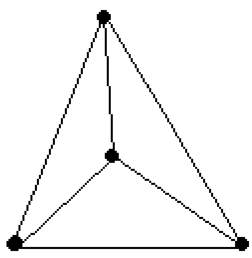


Ответ: не более 10.

42. Можно ли нарисовать графы, изображенные на рисунках, не отрывая карандаш от бумаги и проводя каждое ребро ровно один раз?



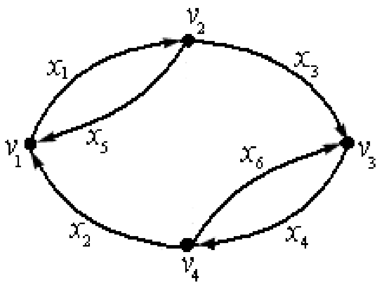
1)



2)

Ответ: 1) можно, так как только две нечетные вершины**; 2)** нельзя, так как нечетных вершин - четыре.

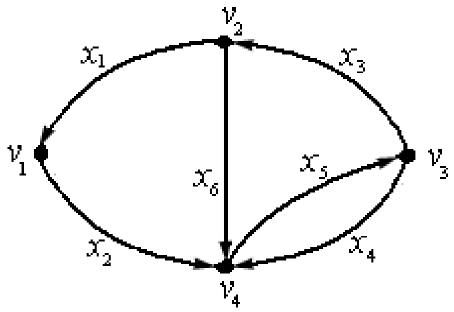
43. Составить матрицу инцидентности данного орграфа:



Решение.

	X1	X2	X3	X4	X5	X6
V1	1	-1	0	0	-1	0
V2	-1	0	1	0	1	0
V3	0	0	-1	1	0	-1
V4	0	1	0	-1	0	1

44. Составить матрицу смежности данного орграфа:

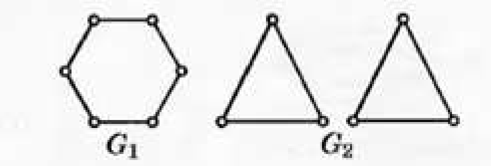


Решение.

	V1	V2	V3	V4
V1	0	0	0	1
V2	1	0	0	1
V3	0	1	0	1
V4	0	0	1	0

45. Может ли так случиться, что в одной компании из шести человек каждый знаком с двумя и только с двумя другими?

Решение. Участников этой компании изобразим вершиной графа, а отношение знакомства между двумя учасниками – ребром. Изобразим графы, которые могут соответствовать такой компании.

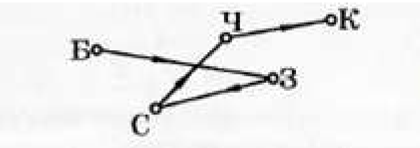


Про граф G_1 говорят, что он связный, так как из каждой вершины по ребрам можно попасть в любую другую. Делаем вывод, что в этом случае каждый через своих знакомых может познакомиться со всеми остальными.

Про граф G_2 говорят, что он несвязный, так как состоит из двух простых циклов. Делаем вывод, что граф соответствует двум компаниям, участники одной из них могут быть не знакомы с участниками другой.

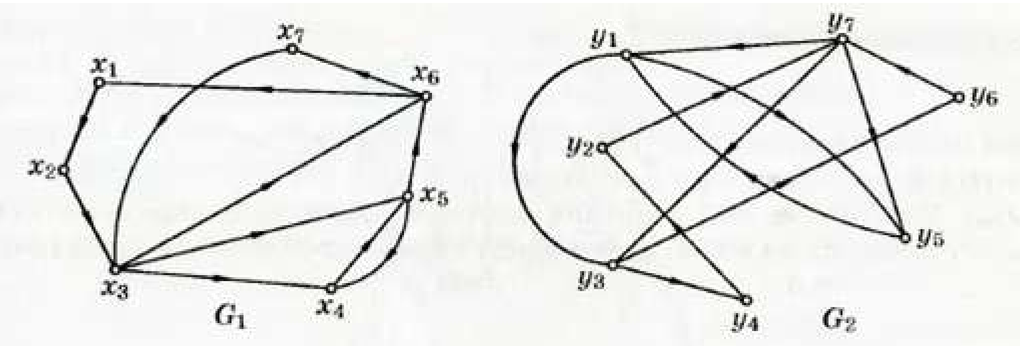
46. Из пункта А в пункт В выехали пять машин одной марки разного цвета: белая, черная, красная, синяя, зеленая. Черная едет впереди синей, зеленая – впереди белой, но позади синей, красная впереди черной. Какая машина едет первой и какая последней?

Решение. Решаем задачу, построив ориентированный граф для отношения f :
« x едет сзади y». На плоскости отметим пять точек, соответствующих каждой машине, и обозначим их первой буквой цвета машины



Анализируя граф, получаем следующий порядок движения: красная, черная, синяя, зеленая, белая.

47. Пусть даны графы $G_1(X, E)$ и $G_2(Y, E)$. Установите, изоморфны ли данные графы



Решение. Для доказательства того, что граф G_1 изоморфен графу G_2 необходим и достаточно выполнение условия: найти такую подстановку, которая граф (переводит в граф G_2.

Запишем элементы $x \in X$ и $y \in Y$ с соответствующими им парами чисе, где первое число — число исходов из вершины, а второе — число заходов вершину. Далее определим частичную подстановку, соединяя вершины x_i и y_i с одинаковыми числами (рис. 5.4).

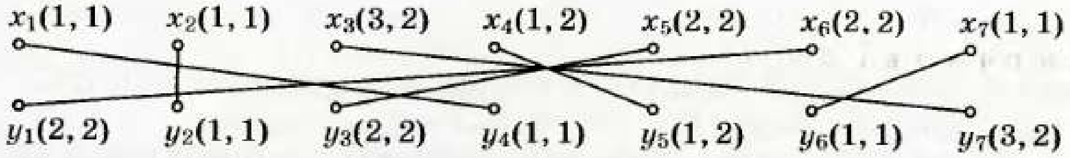


Рис. 5.4

В результате получим подстановку

$$\begin{pmatrix} x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & x_7 \\ y_4 & y_2 & y_7 & y_5 & y_3 & y_1 & y_6 \end{pmatrix}.$$

Следовательно, графы G_1 и G_2 изоморфны.

48. Найдите функции **g** и **h** в рекурсивной формуле для двухместной функции **f(x,y)=x y** , если рекурсия проводиться по переменной **x.**

Решение. Алгоритм решения:

1. Найдите функцию $g(x) = f(0, y)$ (т. к. рекурсия проводится по x).

$$g(y) = f(0, y) = 0 \cdot y = 0.$$

2. Подставить в функцию $f(x, y)$ вместо x значение $x + 1$.

$$f(x+1, y) = (x+1) \cdot y = x \cdot y + y.$$

3. Выразить функцию $f(x+1, v)$ через $f(x, y)$, x, y

$$f(x+1, y) = f(x, y) + y.$$

4. Найти функцию $h(x, y, z)$ из формулы $f(x+1, y) = h(x, y, f(x, y))$, заменив $f(x, y)$ на z.

$$h(x, y, z) = z + y.$$

49. Найдите функции **g** и **h** в рекурсивной формуле для трехместной функции **f(x,y,z) = x y+z,** если рекурсия проводится по переменной **y.**

1. Найти функцию g: $g(x, z) = f(x, 0, z)$ (так как рекурсия проводится по y)

$$g(x, z) = f(x, 0) = x \cdot 0 + z = z.$$

2. Подставить в функцию $f(x, y, z)$ вместо у значение $y + 1$.

$$f(x, y + 1, z) = x(y + 1) + z = xy + z + x.$$

3. Выразить функцию $f(x, y + 1, z)$ через $f(x, y + 1, z)$, x, y, z.

$$f(x, y + 1, z) = (xy + z) + x = f(x, y, z) + x.$$

4. Найти функцию $h(x, y, z, t)$ из формулы

$$f(x, y + 1, z) = h(x, y, z, f(x, y, z)).$$

Заменив $f(x, y, z)$ на t, получим $h(x, y, z, t) = t + x$.

50. Примените оператор минимизации к функции

$$f(x) = \begin{cases} 3x + 2, \text{если x} \neq 2 \\ \text{не определено, если x} = 2 \end{cases}$$

Решение. Для каждого $x_0 \in N$ ищем минимальное решение уравнения $f(y) = x_0$.

Так как множеством значений функции $f(x)$ является множество

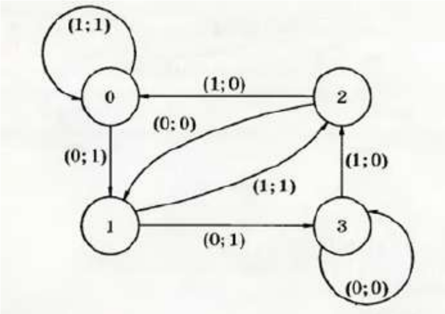
$$\{3n + 2 : \ n \neq 2\} = \{2, 5\} \cup \{11, 14, \ldots, 3n + 2, \ldots\},$$

то уравнение $f(y) = x_0$ имеет решение лишь при $x_0 = 2, 5, 11, 14, \ldots$. Для всякого такого x_0 решение единственное (оно равно $\frac{x_0 - 2}{3}$).

Принимая во внимание, что функция $f(x)$ при $x = 2$ не определена, заключаем: найденные решения, превосходящие 2, т. е. 3, 4, ..., не являются допустимыми в силу определения минимизации. Итак, функция $g(x) = \mu_x f(x)$ определена только при $x = 2$ и $x = 5$; $g(2) = 0$, $g(5) = 1$.

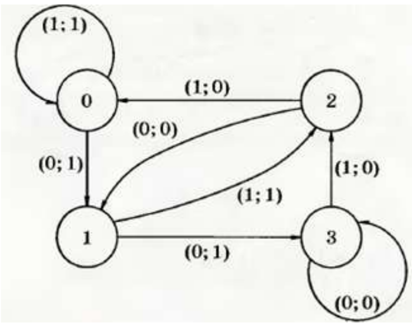
51. Для автомата, заданного таблицей, постройте диаграмму Мура. Задайте этот автомат системой булевых функций

$\alpha \backslash q$	0	1	2	3
0	(1;1)	(3;0)	(2;0)	(2;0)
1	(2;1)	(2;0)	(3;0)	(3;0)

Решение. 

$$\begin{cases} z_1(t+1) = \overline{x}(t) \wedge (z_1(t) \leftrightarrow z_2(t)) \vee z_2(t) \wedge x(t), \\ z_2(t+1) = \overline{x}(t) \wedge z_2(t) \vee x(t) \wedge z_1(t), \\ y_1(t) = \overline{x}(t) \wedge (z_1(t) \vee z_2(t)) \vee x(t) \wedge z_1(t), \\ y_2(t) = \overline{x}(t) \wedge z_1(t) \wedge z_2(t) \vee x(t) \wedge z_1(t) \wedge \overline{z_2}(t); \end{cases}$$

52. Для автомата, заданного диаграммой Мура, выпишите соответственную таблицу и систему булевых функций



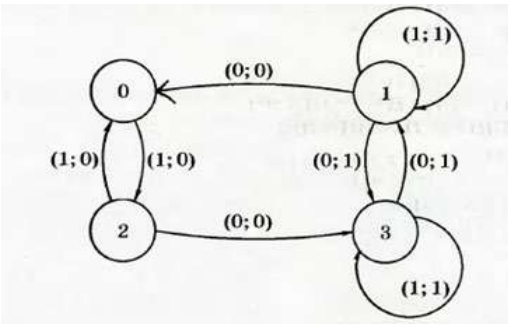
Решение.

$$\begin{cases} z_1(t+1) = z_1(t) \oplus z_2(t) \oplus x(t), \\ z_2(t+1) = z_2(t) \rightarrow x(t), \\ y(t) = z_2(t) \leftrightarrow x(t). \end{cases}$$

$\alpha \backslash q$	0	1	2	3
0	(1;1)	(3;0)	(2;0)	(2;0)
1	(2;1)	(2;0)	(3;0)	(3;0)

53. Для автомата, заданного каноническими уравнениями, постройте диаграмму Мура и соответствующую таблицу $\begin{cases} z(t+1) = x(t), \\ y(t) = x(t) \leftrightarrow z(t) \end{cases}$.

Решение.



$\alpha \backslash q$	0	1	2	3
0	(2;0)	(0;0)	(3;1)	(1;0)
1	(1;0)	(0;0)	(0;0)	(3;0)

7. Критерии оценки:

Оценка ***"отлично"*** выставляется обучающемуся, обнаружившему всестороннее, систематическое и глубокое знание учебно-программного материала, умение свободно выполнять задания, предусмотренные программой, усвоившему основную литературу, рекомендованную программой, взаимосвязь основных понятий дисциплины в их значении для приобретаемой профессии, проявившему творческие способности в понимании, изложении и использовании учебно-программного материала.
Оценка ***"хорошо"*** выставляется обучающему, показавшему полное знание учебно-программного материала, успешно выполняющему предусмотренные в программе задания, усвоившему основную литературу, рекомендованную в программе, показавшему систематический характер знаний по дисциплине и способному к их самостоятельному пополнению и обновлению в ходе дальнейшей учебной работы и профессиональной деятельности.
Оценка ***"удовлетворительно"*** выставляется обучающему, показавшему знания основного учебно-программного материала в объеме, необходимом для дальнейшей учебы и предстоящей работы по специальности, справляющемуся с выполнением заданий, предусмотренных программой, знакомому с основной литературой, рекомендованной программой. Оценка "удовлетворительно" выставляется обучающимся, допустившим погрешности в ответе на теоретические вопросы и при выполнении практических заданий, но обладающим необходимыми знаниями для их устранения под руководством преподавателя.
Оценка "***неудовлетворительно"*** выставляется обучающемуся, обнаружившему пробелы в знаниях основного учебно-программного материала, допустившему принципиальные ошибки в выполнении предусмотренных программой заданий.

8. Перечень материалов, оборудования и информационных источников, используемых при проведении промежуточной аттестации

При проведении промежуточной аттестации обучающиеся могут пользоваться конспектом лекций, справочными материалами.

9. Приложение: перечень вопросов к дифзачету

Теоретические задания:

1. Взаимосвязь дискретной математики с другими дисциплинами. Практические проблемы, изучаемые методами дискретной математики

2. Составные высказывания. Простейшие связки. Логические отношения.

3. Варианты импликации.

4. Основные законы, определяющие свойства логических операций.

5. Булевы функции.

6. Свойства элементарных булевых функций.

7. Дизъюнктивные и конъюнктивные нормальные формы алгебры высказываний.

8. Совершенная дизъюнктивная и совершенная конъюнктивная нормальные формы.

9. Многочлены Жегалкина.

10. Специальные классы булевых функций: функции, сохраняющие единицу, функции, сохраняющие нуль, самодвойственные функции,линейные функции, монотонные функции. Теорема Поста о функциональной полноте.

11. Понятие множества. Способы задания множества. Подмножества. Операции над множествами.

12. Соотношения между множествами и составными высказываниями.

13. Соотношение между высказываниями и соответствующими им множествами истинности.

14. Абстрактные законы операций над множествами.

15. Кортежи и декартово произведение множеств. Степень множества.

16. Бинарные отношения в множестве и их свойства.

17. Отношения строгого и нестрогого порядка.

18. Отображение множеств. Функции.

19. Определенность и неопределенность функций. Композиция отображений.

20. Метод математической индукции. База индукции. Индукционный переход. Полная и неполная индукция.

21. Основные правила комбинаторики. Методы алгоритмического перечисления (генерации) основных комбинаторных объектов: перестановка, сочетание, размещение.

22. Комбинация элементов с повторениями. Бином Ньютона.

23. Предикаты. Применение предикатов в алгебре.

24. Булева алгебра предикатов.

25. Кванторы. Формулы логики предикатов.

26. Равносильные формулы логики предикатов. Приведенные и нормальные формы в логике предикатов.

27. Исчисления предикатов.

28. Основные понятия теории графов. Степень вершины. Маршрут, цепи, циклы. Связность графа.

29. Ориентированные графы.

30. Изоморфизм графов.

31. Плоские графы. Операции над графами.

32. Способы задания графов. Некоторые типы графов.

33. Сети. Сетевые модели представления информации. Применение графов и сетей.

34. Вычислимые функции и алгоритмы.

35. Рекурсивные функции.

36. Нормальный алгоритм Маркова.

37. Машины Тьюринга.

38. Понятия конечного автомата. Определения и способы задания конечного автомата.

39. Примеры конечных автоматов.

40. Канонические уравнения автомата.

Практические задания:

1. Докажите тождественную истинность формулы $\overline{X} \rightarrow (\mathbf{X} \rightarrow \mathbf{У})$.

2. С помощью таблиц истинности проверить, являются ли эквивалентными высказывания: $\mathbf{f_1 = X \wedge (Y \rightarrow Z)}$ **и** $\mathbf{f_2 = (\overline{X} \wedge Y) \vee (X \wedge Z)}$.

3. Определите для каждого из следующих высказываний, будет ли оно логически истинным, противоречивым: ни тем, ни другим.

а) $X \leftrightarrow X$, б)$X \leftrightarrow \overline{X}$, в)$(X \vee У) \leftrightarrow (X \wedge У)$, г)$(X \rightarrow \overline{У}) \rightarrow (У \rightarrow \overline{X})$,
д)$(X \rightarrow Y) \wedge (Y \rightarrow Z) \wedge \overline{(X \rightarrow Z)}$, е)$(X \rightarrow У) \rightarrow X$, ж)$((X \rightarrow У) \rightarrow X$

4. Доказать закон отрицания конъюнкции ($\overline{A \wedge B} \leftrightarrow \overline{A} \vee \overline{B}$)
5. Найти значение $\overline{A \wedge B} \leftrightarrow \overline{A} \vee \overline{B}$ и убедиться, что при всех значениях А и В - это истинное значение.
6. С помощью основных равносильностей доказать закон обобщенного склеивания $x\,y \vee \bar{x}\,z = x\,y \vee \bar{x}\,z \vee y\,z$.

7. Составьте таблицу истинности булевой функции трех переменных $f(x_1, x_2, x_3) = x_1 \oplus x_2 \rightarrow \overline{x3} \vee x_1 \,|(\overline{x2} \wedge \overline{x1})$ и найдите ее двоичный набор.

8. Докажите эквивалентность функции: $f_{(x, y, z)} = x \wedge (x \vee z) \wedge (y \vee z)$ и $f_{(x, y, z)} = (x \wedge y) \vee (x \wedge z)$.

9. Найдите СДНФ и СКНФ функции **$f_{(x, y, z)}$**, заданной следующей таблицей истинности:

x_1	x_2	x_3	$f(x_1, x_2, x_3)$
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

10. Опрос 100 студентов выявил следующие данные о числе студентов, изучающих различные иностранные языки: английский – 28; немецкий - 30; французский – 42, английский и немецкий – 8; английский и французский – 10; немецкий и французский – 5; все три языка – 3.

1) Сколько студентов не изучают ни одного иностранного языка?

2) Сколько студентов изучают один французский язык?

3) Сколько студентов изучают немецкий язык в том и только в том случае, если они изучают французский язык?

Решение. Нарисовать диаграмму Эйлера-Венна в виде трех кругов, обозначающих множество студентов, изучающих соответственно французский, немецкий и английский языки. В каждую из 8-ми областей вписать данные, используя приведенные цифры. Начинать с конца списка и двигаться к началу.

Ответ: 1) 20; 2) 30; 3) 38.

11. Изобразите с помощью диаграмм Эйлера-Венна множества:
1) $A \subset B$ и $B \subset C$; 2) $A \subset B$; $B \subset C$ и $A \setminus B = \emptyset$; 3)$A \subset B$; $B \subset C$ и $C = A \cup B$;

4) $A \subset B$; $B \subset C$ и $A \cap B \neq \emptyset$; 5)$(A \setminus B) \cap (B \setminus A)$.

12. Воспользовавшись диаграммой Эйлера-Венна, определите, какие из следующих высказываний истинны: а)$\mathbf{X} \curlyvee \overline{\boldsymbol{X}}$; **б)** $\mathbf{X} \curlywedge \overline{\boldsymbol{X}}$; **в)** $\mathbf{X} \curlyvee (\overline{\boldsymbol{X}} \wedge \boldsymbol{Y})$;
г) $\mathbf{X} \to (\mathbf{Y} \to \mathbf{X})$; **д)**$\mathbf{X} \wedge \overline{(\boldsymbol{Y} \to \boldsymbol{X})}$.

13. Пусть $A = \{1, 2\}$. Выписать все элементы декартова произведения $A \times A$.

14. Рассмотрим два множества $A = \{a, b, c, d, e, f, g, h\}$ и $B = \{1, 2, 3, 4, 5, 6, 7, 8\}$.

15. Проверьте на линейность функцию $f(x_1, x_2, x_3)$, если ее двоичный набор F= 11100001.

16. Найдите правую и левую области отношения $R = \{\langle 1, 5\rangle ; \langle 1, 6\rangle ; \langle 1, 7\rangle\}$.

17. Если $A = \{2, 3,4, 5, 6, 7, 8\}$, запишите бинарное отношение $R = \{\langle x, y\rangle : x, y \in A$,

x делит y, и $x \le 3\}$.

18. Являются ли следующие отношения функциями:

1) $\{\langle 1, 2\rangle ; \langle 2, 3\rangle ; \langle 3, 2\rangle\}$; 2) $\{\langle 1, 2\rangle ; \langle 1, 3\rangle ; \langle 2, 3\rangle\}$;
3) $\{x, x^2 - 2x - 3 : x \in R\}$?

19. Футбольный мяч сшит из 32 лоскутов: белых шестиугольников и чёрных пятиугольников. Каждый чёрный лоскут граничит только с белыми, а каждый белый – с тремя чёрными и тремя белыми. Сколько лоскутов белого цвета?

20. Из 4 первокурсников, 5 второкурсников и 6 третьекурсников надо выбрать 3 студента на конференцию. Сколькими способами можно осуществить этот выбор, если среди выбранных должны быть студенты разных курсов?

21. Сколько можно составить четырехзначных чисел так, чтобы любые две соседние цифры были различны?

22. Сколькими способами можно рассадить 5 человек за круглым столом (рассматривается только расположение сидящих относительно друг друга)?

23. Сколькими способами можно распределить 15 выпускников по 3 районам, если в одном из них имеется 8, в другом - 5 и в третьем – 2 вакантных места?

24. Известно, что 7 студентов сдали экзамен по дискретной математике на хорошо и отлично. Сколькими способами могли быть поставлены им оценки?

25. Группа студентов изучает 10 различных дисциплин. Сколькими способами можно составить расписание занятий в понедельник, если в этот день должно быть 4 разных занятия?

26. Из 60 вопросов, входящих в экзаменационные билеты, студент знает 50. Найти вероятность того, что среди трех наугад выбранных вопросов студент знает: а) все вопросы, б) два вопроса.

27. Во взводе три сержанта и 30 солдат. Сколькими способами можно выделить одного сержанта и трех солдат для патрулирования?

28. В барабане револьвера 7 гнезд, из них в 5 заложены патроны. Барабан приводится во вращение, потом нажимается спусковой курок. Какова вероятность того, что, повторив такой опыт 2 раза подряд: а) револьвер оба раза не выстрелит, б) оба раза револьвер выстрелит.

29. Решить уравнение: $5C_x^3 = C_{x+2}^4$

30. Решить уравнение: $C_{x-3}^2 = 21$

31. Решить уравнение: $A_x^3 = \frac{1}{20}A_x^4$

32. Решить уравнение: $C_x^3 + C_x^2 = 15(x-1)$

33. Решить уравнение: $C_x^3 = \frac{5x\,(x-3)}{4}$

34. Записать предложение: «прямая a параллельна прямой b» с помощью предиката.

35. Записать с помощью предиката: «Аксиома: через две различные точки проходит единственная прямая» (Если две точки принадлежат двум прямым, то эти прямые совпадают).

36. Между планетами введено космическое сообщение по следующим маршрутам: З-К, П-В, З-П, П-К, К-В, У-М, М-С, С-Ю, Ю-М, М-У. Можно ли добраться с З до М?

37. Аркадий, Борис. Владимир, Григорий и Дмитрий при встрече обменялись рукопожатиями (каждый пожал руку каждому по одному разу). Сколько всего рукопожатий было сделано?

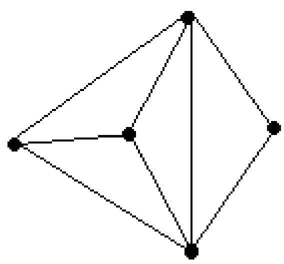
38. К XVIII веку через реку, на которой стоял город Кенигсберг (ныне Калининград), было построено 7 мостов, которые связывали с берегами и друг с другом два острова, расположенные в пределах города.
Задача заключается в следующем: нужно пройти (если это возможно) по всем семи мостам так, чтобы на каждом из них побывать лишь по одному разу и вернуться к тому месту, откуда начал маршрут.

39. В трех различных домах живут три поссорившиеся между собой соседа. Недалеко от их домов имеются три колодца. Можно ли от каждого дома проложить к каждому из колодцев тропинку так, чтобы никакие две из них не пересекались?

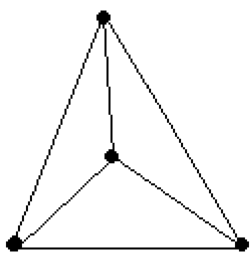
40. В городе Н от каждой площади отходит ровно пять улиц, соединяющих площади. Докажите, что число площадей чётно, а число улиц кратно пяти.

41. В государстве система авиалиний устроена таких образом, что любой город соединён авиалиниями не более чем с тремя другими и из любого города в любой другой можно проехать, сделав не более одной пересадки. Какое максимальное число городов может быть в этом государстве?

42. Можно ли нарисовать графы, изображенные на рисунках, не отрывая карандаш от бумаги и проводя каждое ребро ровно один раз?

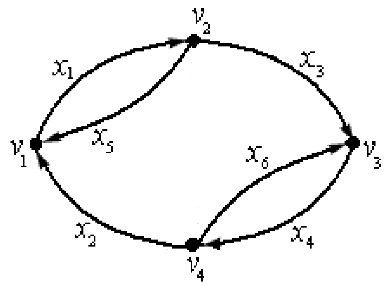


1)

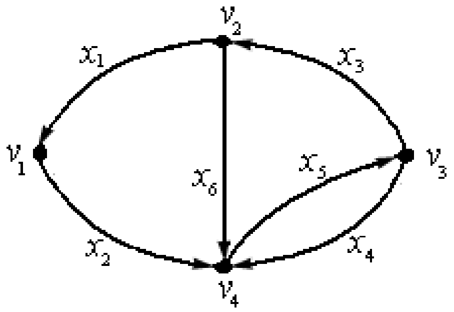


2)

43. Составить матрицу инцидентности данного орграфа:



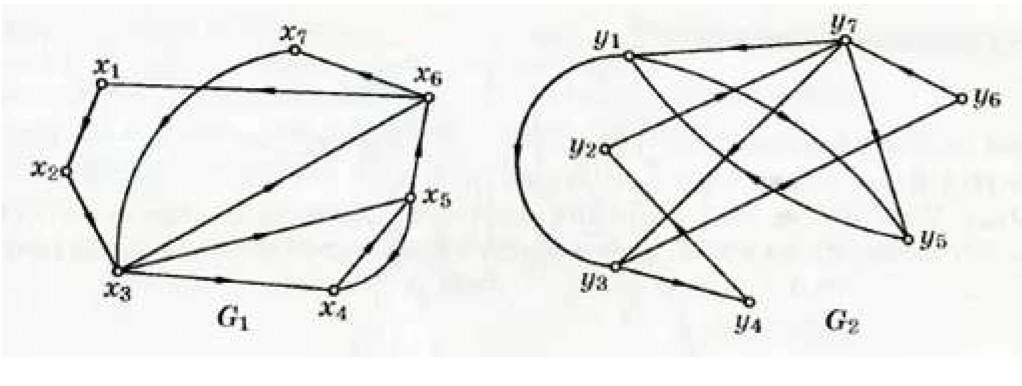
44. Составить матрицу смежности данного орграфа:



45. Может ли так случиться, что в одной компании из шести человек каждый знаком с двумя и только с двумя другими?

46. Из пункта А в пункт В выехали пять машин одной марки разного цвета: белая, черная, красная, синяя, зеленая. Черная едет впереди синей, зеленая – впереди белой, но позади синей, красная впереди черной. Какая машина едет первой и какая последней?

47. Пусть даны графы $G_1(X, E)$ и $G_2(Y, E)$. Установите, изоморфны ли данные графы



48. Найдите функции **g** и **h** в рекурсивной формуле для двухместной функции **f(x,y)=x y** , если рекурсия проводиться по переменной **x.**

49. Найдите функции **g** и **h** в рекурсивной формуле для трехместной функции **f(x,y,z) = x y+z,** если рекурсия проводится по переменной **y.**

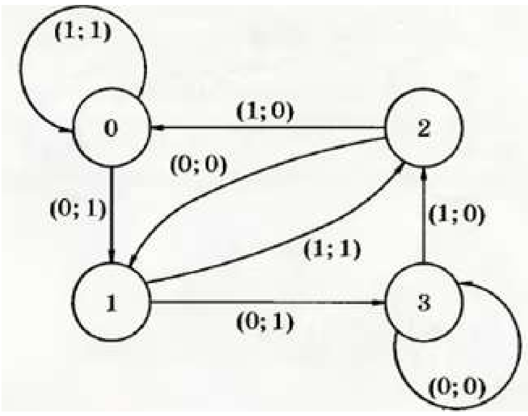
50. Примените оператор минимизации к функции

$$f(x) = \begin{cases} 3x + 2, \text{если } x \neq 2 \\ \text{не определено, если } x = 2 \end{cases}$$

51. Для автомата, заданного таблицей, постройте диаграмму Мура. Задайте этот автомат системой булевых функций

α \ q	0	1	2	3
0	(1;1)	(3;0)	(2;0)	(2;0)
1	(2;1)	(2;0)	(3;0)	(3;0)

52. Для автомата, заданного диаграммой Мура, выпишите соответственную таблицу и систему булевых функций



53. Для автомата, заданного каноническими уравнениями, постройте диаграмму Мура и соответствующую таблицу $\begin{cases} z(t+1) = x(t), \\ y(t) = x(t) \leftrightarrow z(t) \end{cases}$.