Оглавление

[1.Кольцо (коммутативное ассоциативное кольцо с единицей). 4](#_Toc516743709)

[2.Прямое произведение колец. 4](#_Toc516743710)

[3.Изоморфизм колец. 4](#_Toc516743711)

[*4.* Идемпотентные элементы. Критерий разложимости кольца 4](#_Toc516743712)

[в прямое произведение колец. 4](#_Toc516743713)

[5.БУЛЕВО КОЛЬЦО 4](#_Toc516743714)

[6. Теорема о строении конечных булевых колец. 5](#_Toc516743715)

[7. Алгебра логики (булева алгебра) 5](#_Toc516743716)

[8. Теорема о соответствии между булевыми кольцами и буле- 5](#_Toc516743717)

[выми алгебрами. Строение конечных булевых алгебр. 5](#_Toc516743718)

[9. Кольцо многочленов от нескольких переменных над комму- 6](#_Toc516743719)

[тативным кольцом и над полем. 6](#_Toc516743720)

[10. Деление с остатком в кольце многочленов от одной переменной. 7](#_Toc516743721)

[11. Теорема Безу. 8](#_Toc516743722)

[14. Интерполяционный многочлен Лагранжа. 8](#_Toc516743723)

[15. Теорема: любую функцию от нескольких переменных в поле 9](#_Toc516743724)

[из *q* элементов можно однозначно задать многочленом, в 9](#_Toc516743725)

[который все переменные входят в степенях, меньших *q*. 9](#_Toc516743726)

[16.Булева функция. 10](#_Toc516743727)

[17.Носитель булевой функции. 12](#_Toc516743728)

[18.Дизъюнктивная Нормальная Форма (ДНФ). 12](#_Toc516743729)

[19.Совершенная Дизъюнктивная Нормальная Форма (СДНФ). 13](#_Toc516743730)

[20.Многочлен Жегалкина. 16](#_Toc516743731)

[21.Булев куб; его грани и способы их задания. 18](#_Toc516743732)

[22.Сокращенная дизъюнктивная нормальная форма. 22](#_Toc516743733)

[23.Задача о нахождении тупикового покрытия; тупиковое покрытие. 26](#_Toc516743734)

[24.«Максимальный интервал» — максимальная (по включению) грань куба, содержащаяся в носителе булевой функции. 32](#_Toc516743735)

[25.Матрица инциденции покрытия 33](#_Toc516743736)

[26.Ядро покрытия; ядровая ДНФ. 39](#_Toc516743737)

[27. Задача о нахождении тупиковых ДНФ булевой функции. 40](#_Toc516743738)

[28. Схема из функциональных элементов; реализация булевой функции схемой из функциональных элементов 40](#_Toc516743739)

[29. Полная система функций 40](#_Toc516743740)

[30. Замкнутый класс функций 41](#_Toc516743741)

[31. Монотонная булева функция. Критерий монотонности в терминах сокращенной ДНФ. 41](#_Toc516743742)

[32. Линейная булева функция. Необходимое условие линейности ( равное число значений 0 и 1 ) 41](#_Toc516743743)

[33. Булева функция, двойственная к данной. Построение таблицы значений двойственной функции. 42](#_Toc516743744)

[34. Самодвойственная булева функция. Как по таблице значений проверить самодвойственность? 42](#_Toc516743745)

[35. Композиция булевых функций 43](#_Toc516743746)

[36. Пять основных замкнутых классов 44](#_Toc516743747)

[37. Теорема Поста 45](#_Toc516743748)

[38.Граф 45](#_Toc516743749)

[Способы задания графа 45](#_Toc516743750)

[Матрица смежности 48](#_Toc516743751)

[Матрица инцидентности 48](#_Toc516743752)

[39.Автоморфизм графа 49](#_Toc516743753)

[40.Изоморфизм графов 50](#_Toc516743754)

[Гомеоморфизм графов 50](#_Toc516743755)

[Гомотопическая эквивалентность графов 50](#_Toc516743756)

[41.Цикломатическое число 52](#_Toc516743757)

[42.Путь в графе 52](#_Toc516743758)

[Простой путь 52](#_Toc516743759)

[Простой замкнутый путь??? 52](#_Toc516743760)

[43.Поиск кратчайшего пути 52](#_Toc516743761)

[Алгоритм Дейкстры 52](#_Toc516743762)

[Инициализация 53](#_Toc516743763)

[Шаг алгоритма 53](#_Toc516743764)

[Поиск в ширину 53](#_Toc516743765)

[Алгоритм по шагам 54](#_Toc516743766)

[44.Связность графа 54](#_Toc516743767)

[Компонента связности 54](#_Toc516743768)

[Алгоритм проверки связности графа 55](#_Toc516743769)

[Обход в глубину 55](#_Toc516743770)

[Алгоритм проверки связности 55](#_Toc516743771)

[45.Критерий эйлеровости связного графа 55](#_Toc516743772)

[Теорема 1 55](#_Toc516743773)

[Теорема 2 55](#_Toc516743774)

[Теорема 3 56](#_Toc516743775)

[Утверждения 56](#_Toc516743776)

[46.Двудольные графы 56](#_Toc516743777)

[Теорема Кёнига 57](#_Toc516743778)

[Доказательство 57](#_Toc516743779)

[Следствия 57](#_Toc516743780)

[**Алгоритм проверки графа на двудольность, используя обход в глубину** 57](#_Toc516743781)

[**Алгоритм проверки графа на двудольность, используя обход в ширину** 58](#_Toc516743782)

[47.Числа реберной и вершинной связности графа 58](#_Toc516743783)

[Теорема 1 58](#_Toc516743784)

[Доказательство 58](#_Toc516743785)

[Теорема 2 59](#_Toc516743786)

[Доказательство 59](#_Toc516743787)

[48. Дерево — связный граф с нулевым цикломатическим числом. 59](#_Toc516743788)

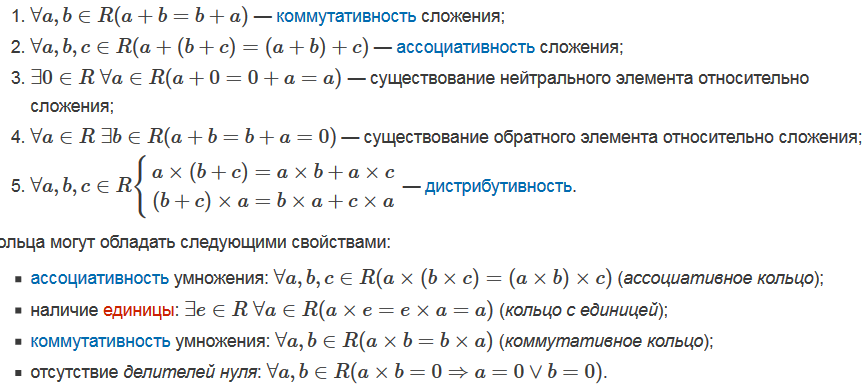
[49. Свойства деревьев 60](#_Toc516743789)

[50. Валентность вершины графа. Паспорт графа. Теорема о рукопожатиях. Каким может быть паспорт связного графа? 60](#_Toc516743790)

[51. Остовное дерево связного графа. Жадный алгоритм. 61](#_Toc516743791)

[52. Определение потока на графе. Условия Кирхгофа — линейные уравнения. 61](#_Toc516743792)

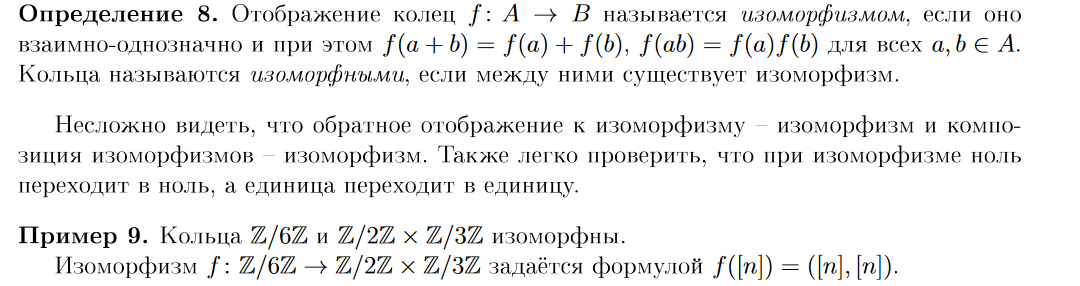
# 1.Кольцо (коммутативное ассоциативное кольцо с единицей).

**Koльцо** **-** [*множество*](http://ru.math.wikia.com/wiki/%D0%9C%D0%BD%D0%BE%D0%B6%D0%B5%D1%81%D1%82%D0%B2%D0%BE)*, на котором заданы две операции, «сложение» и «умножение», со свойствами*

# 2.Прямое произведение колец.

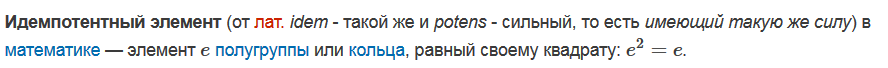
**Прямым произведение колец А и В**называется теоретико-множественное (декартово) произведение АхВ с покомпонентными операциями. Оно также называется кольцом.

# 3.Изоморфизм колец.

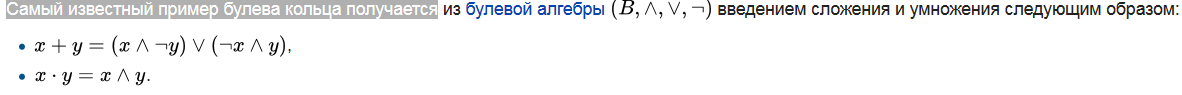


# *4.* Идемпотентные элементы. Критерий разложимости кольца

# в прямое произведение колец.



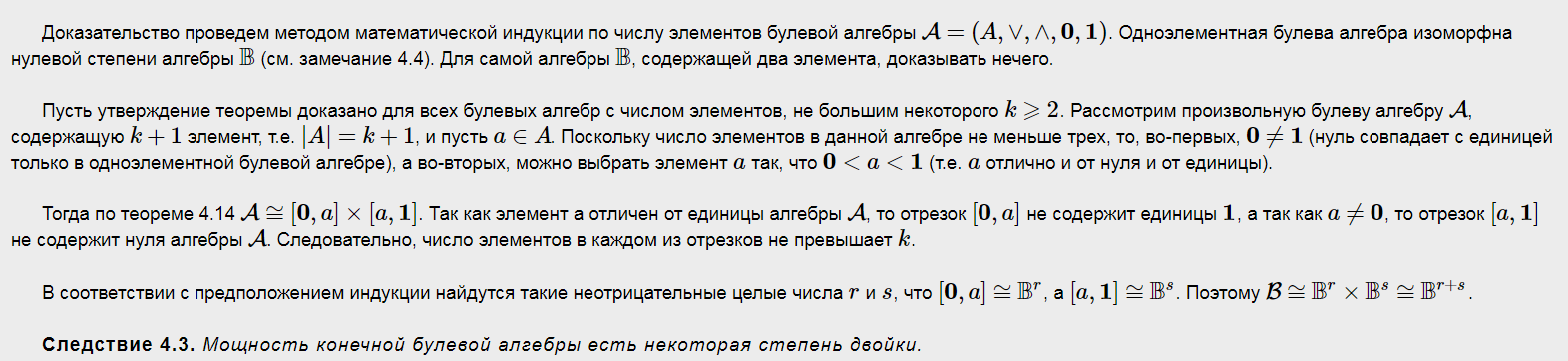
5.БУЛЕВО КОЛЬЦО - ассоциативное кольцо K, все элементы к-рого идемпотентны, т. е. х2 = х для любого х ∈ К. Любое Б. к. K ≠ 0 коммутативно и является подпрямой суммой полей Z2 из двух элементов. При этом х + х = 0 для всех х ∈ К. конечное Б. к. K ≠ 0 является прямой суммой полей Z2 и потому имеет единицу.



# 6. Теорема о строении конечных булевых колец.

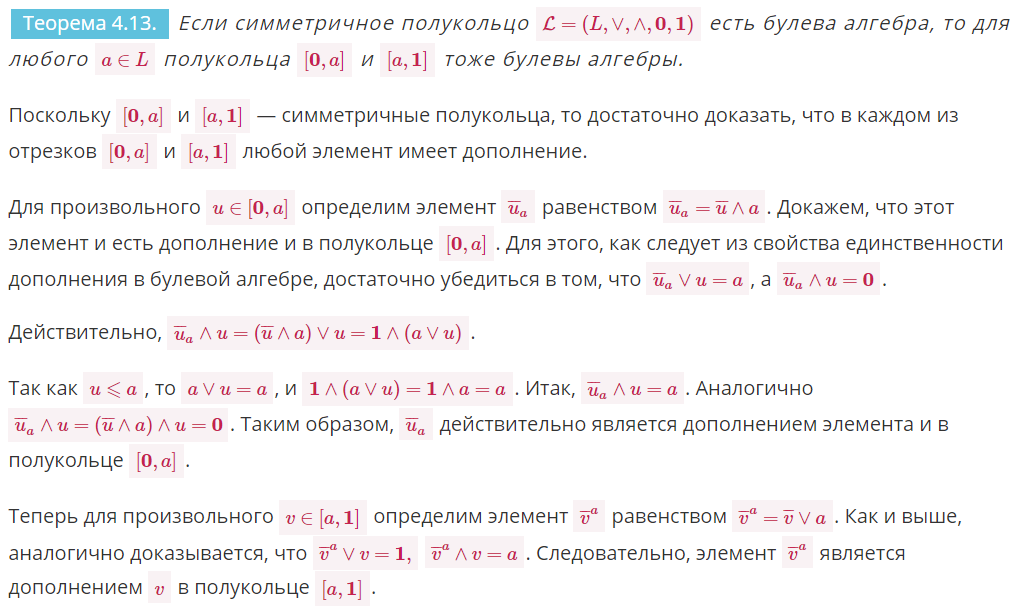
**Теорема 4.15.** Любая конечная булева алгебра изоморфна булевой алгебре

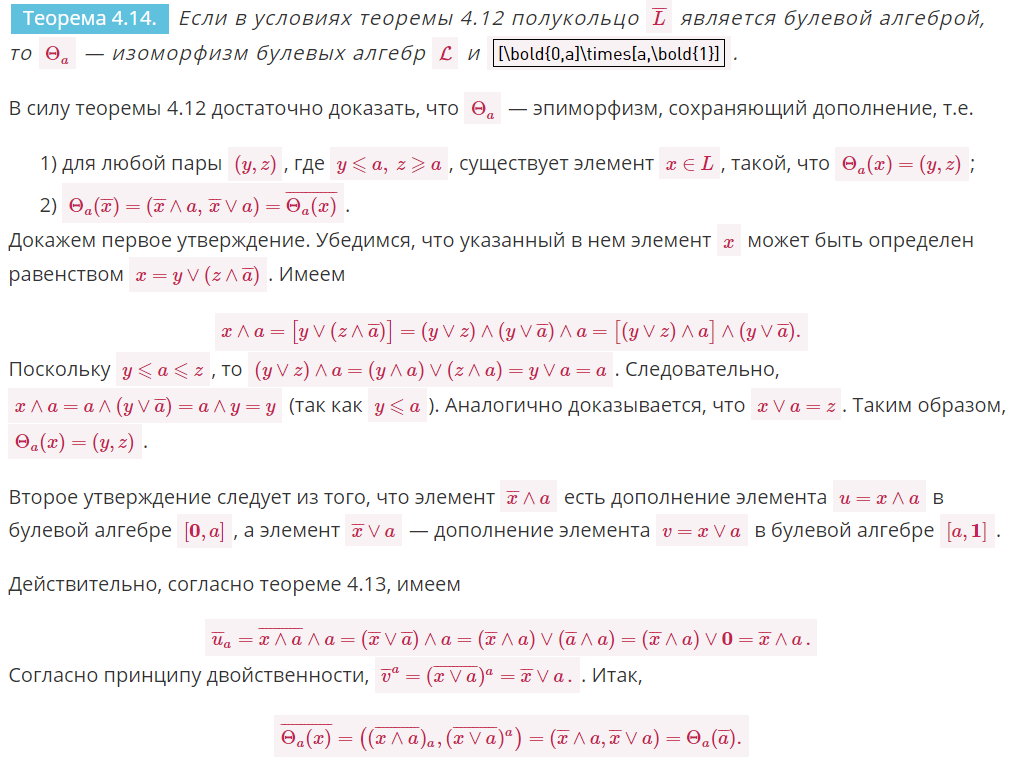
Bn для некоторого n.



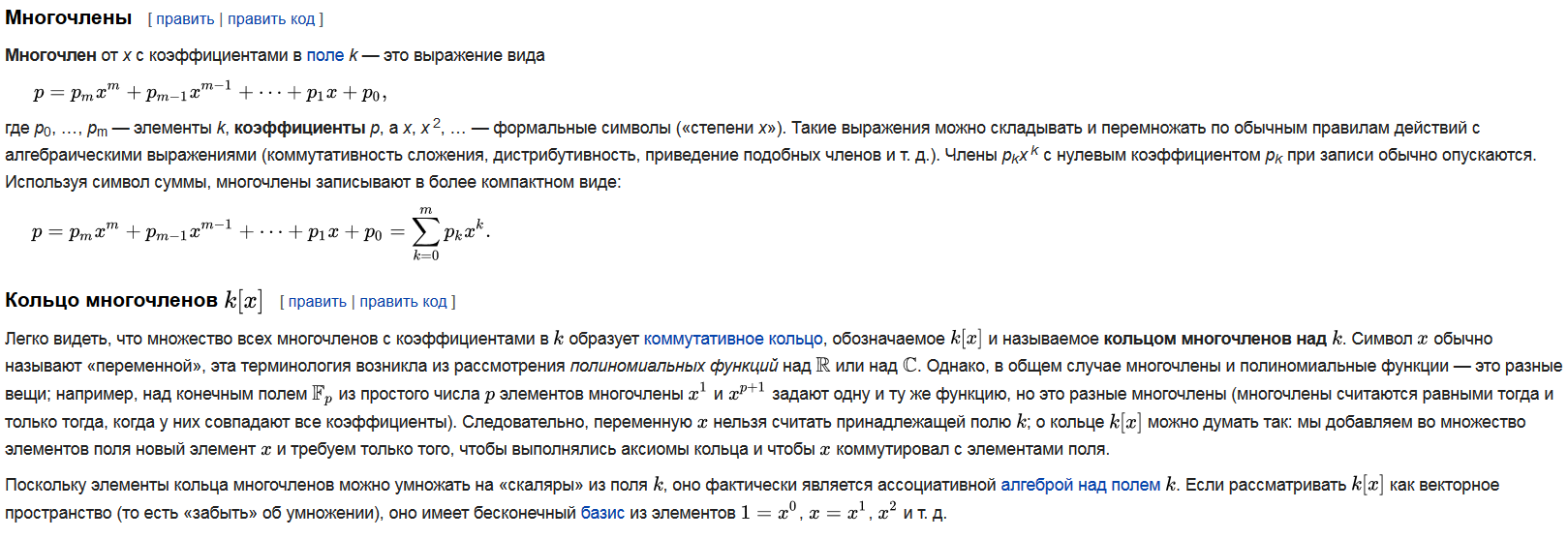
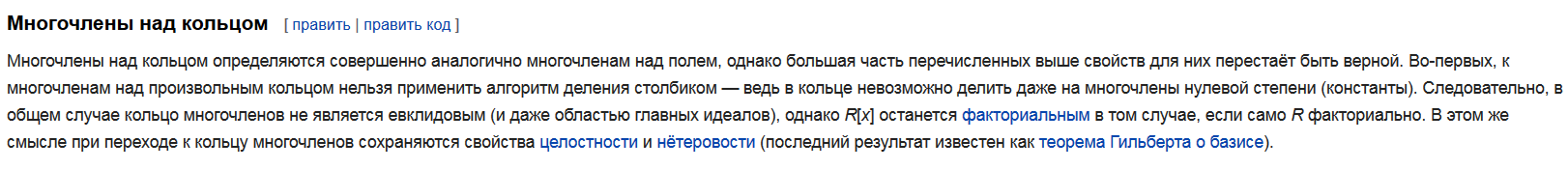
7. Алгебра логики (булева алгебра) — это раздел математики, изучающий высказывания, рассматриваемые со стороны их логических значений (истинности или ложности) и логических операций над ними. Алгебра логики позволяет закодировать любые утверждения, истинность или ложность которых нужно доказать, а затем манипулировать ими подобно обычным числам в математике.

# 8. Теорема о соответствии между булевыми кольцами и булевыми алгебрами. Строение конечных булевых алгебр.

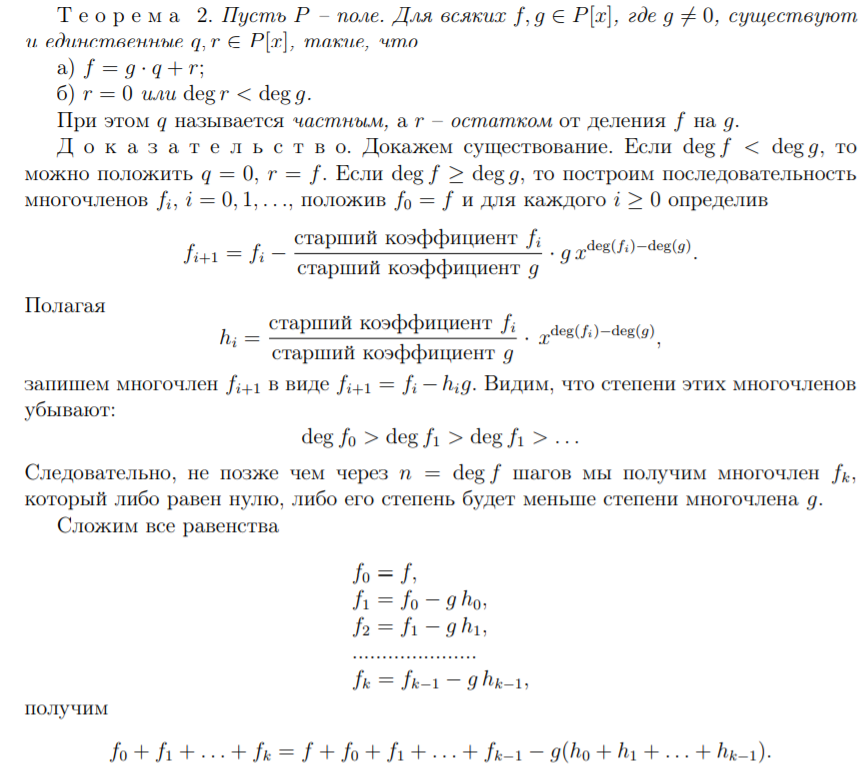


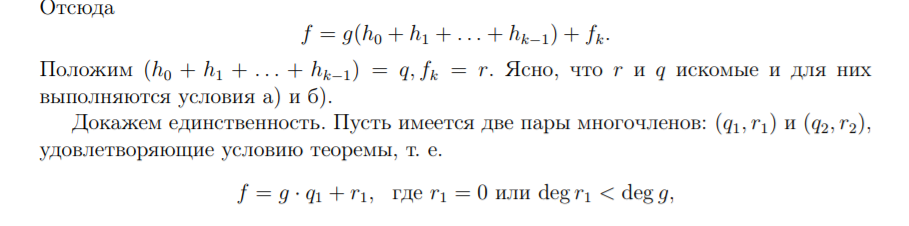


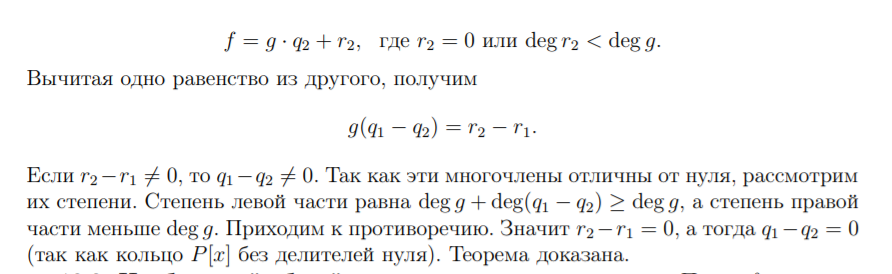
# 9. Кольцо многочленов от нескольких переменных над коммутативным кольцом и над полем.



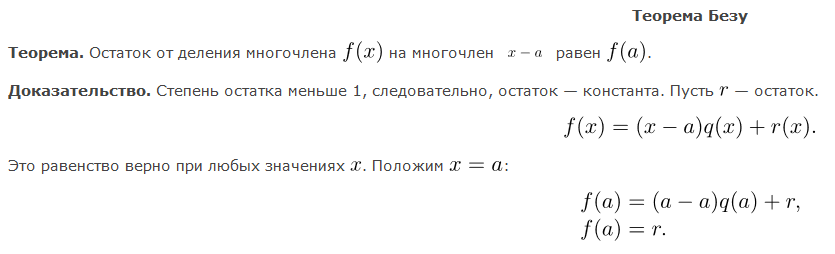
# 10. Деление с остатком в кольце многочленов от одной переменной.







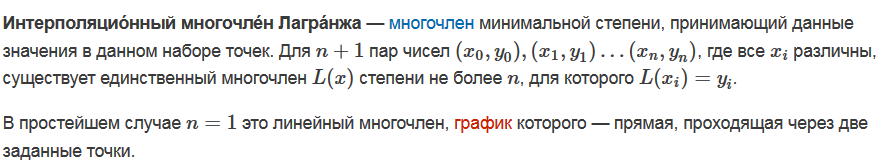
# 11. Теорема Безу.

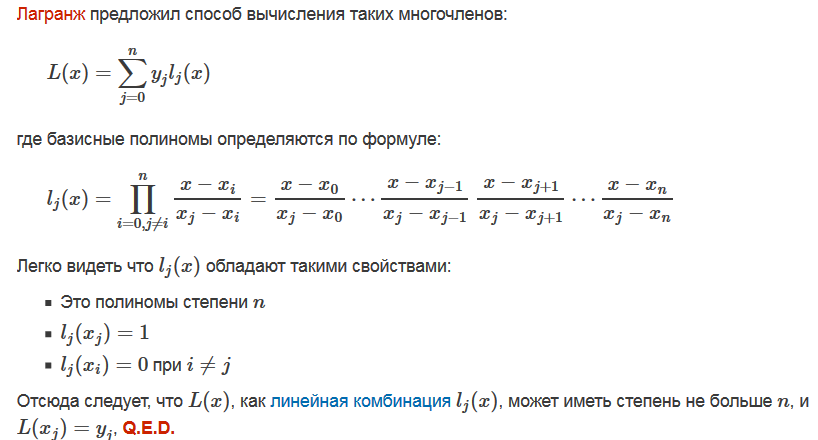


12. GoogelException: этой хуйни не существует

13. GoogelException: этой хуйни не существует

# 14. Интерполяционный многочлен Лагранжа.



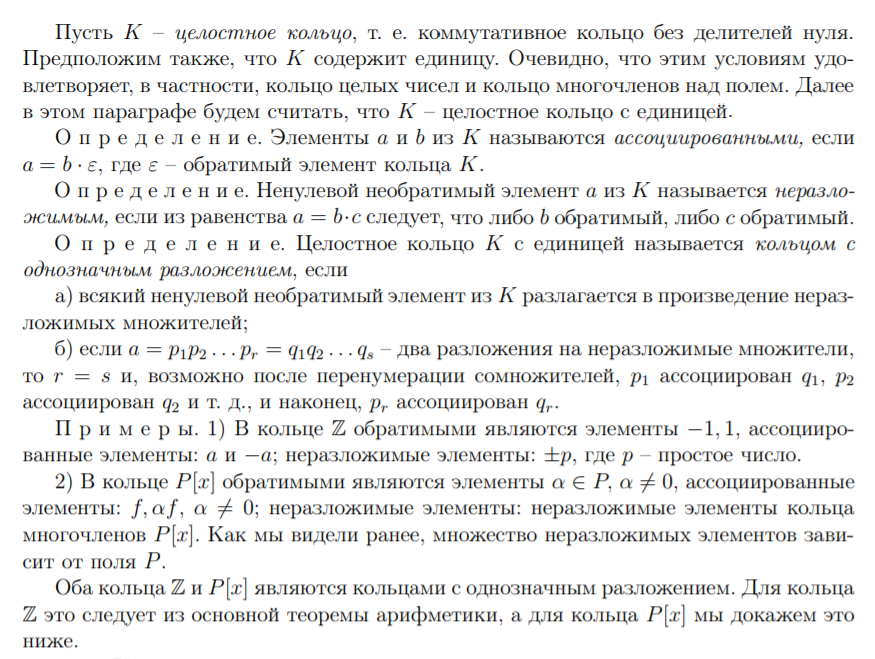


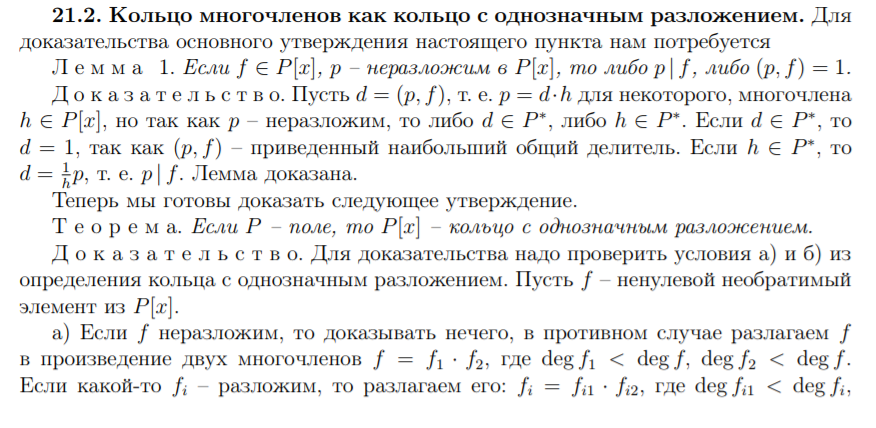
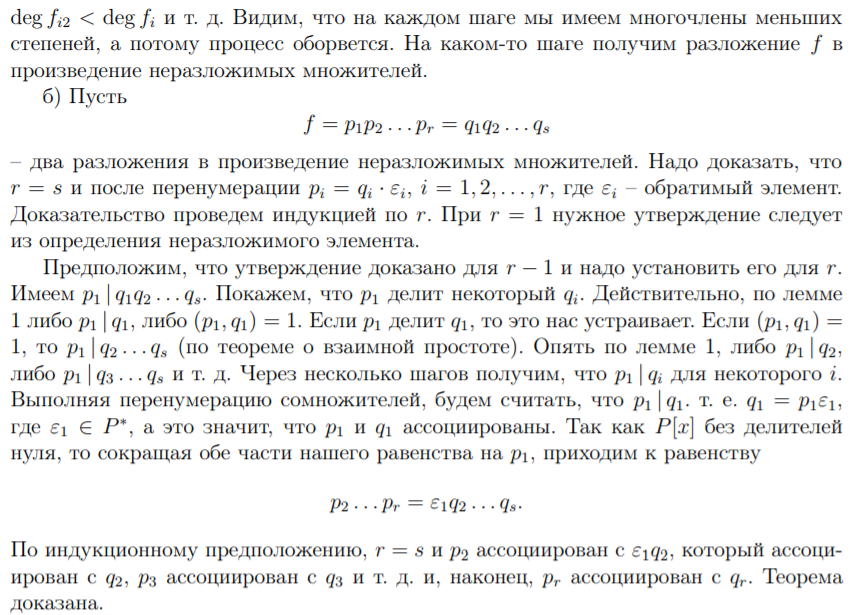
Полиномы Лагранжа используются для интерполяции, а также для численного интегрирования.

# 15. Теорема: любую функцию от нескольких переменных в поле

# из *q* элементов можно однозначно задать многочленом, в

# который все переменные входят в степенях, меньших *q*.

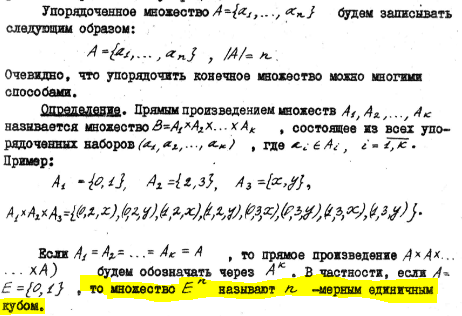


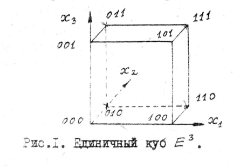
# 16.Булева функция.

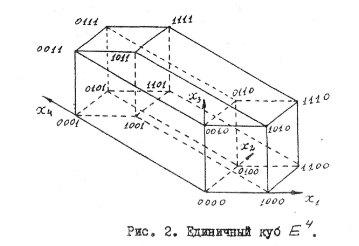
Ответ.

Для того, чтобы понять данное определение булевой функции, необходимо понимать, что такое *n-мерный единичный булев куб*:



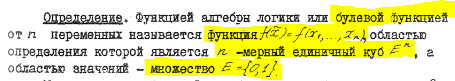
Примеры единичного булева куба:





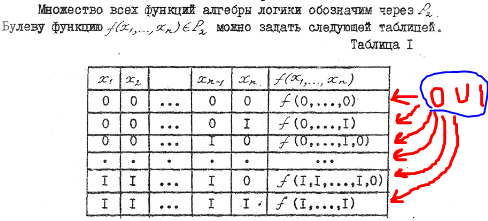
Более подробно про единичный булев куб поговорим в соответствующем билете.

Итак, определение булевой функции:



# 17.Носитель булевой функции.

Ответ.

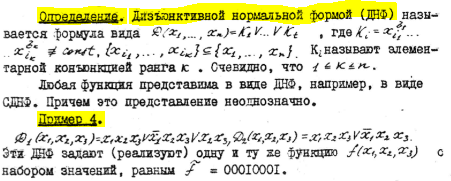


Наборы, на которых значение функции равно 1, называют единичными наборами. Множество всех единичных наборов данной функции - единичным множеством или **носителем булевой функции** .

# 18.Дизъюнктивная Нормальная Форма (ДНФ).

Ответ.

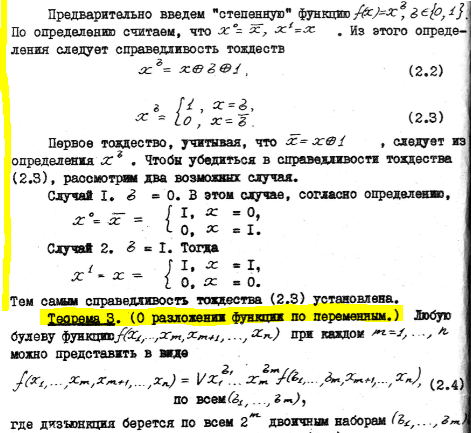
**Определение Дизъюнктивной Нормальной Формы (ДНФ).**



# 19.Совершенная Дизъюнктивная Нормальная Форма (СДНФ).

Ответ.

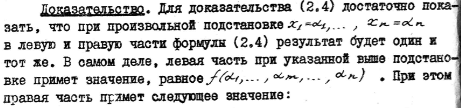
Для того, чтобы хорошо понимать, что такое **Совершенная Дизъюнктивная Нормальная Форма (ДНФ)**, нужно знать *Теорему о разложении булевой функции по переменным*:

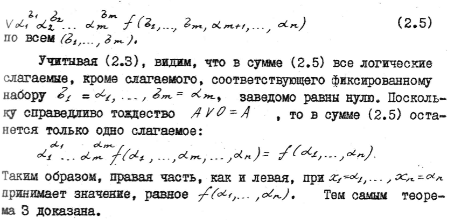


Для лучшего понимания этой теоремы, предлагаю посмотреть такое объяснение:

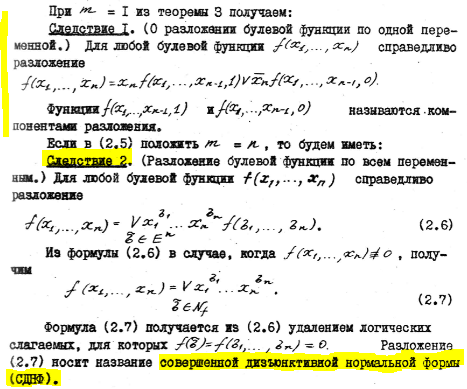


Предлагаемое доказательство (по сути, то же самое, что и на прошлой картинке, но только более строго):

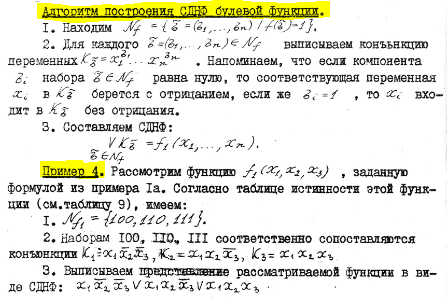




Одним из следствий этой теоремы является определение **Совершенной Дизъюнктивной Нормальной Формы (СДНФ)**:



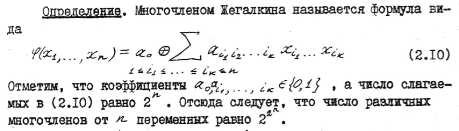
*Для практических нужд:*



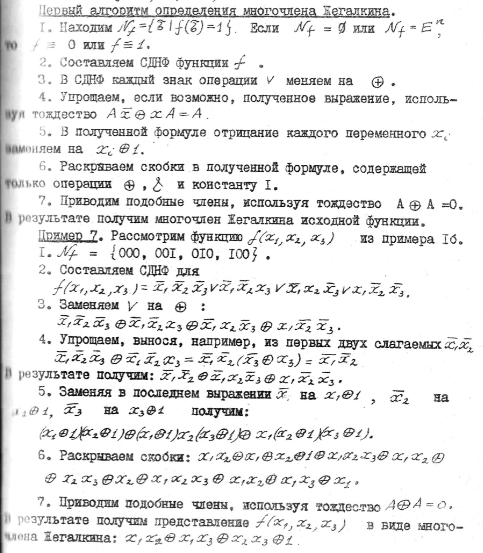
# 20.Многочлен Жегалкина.

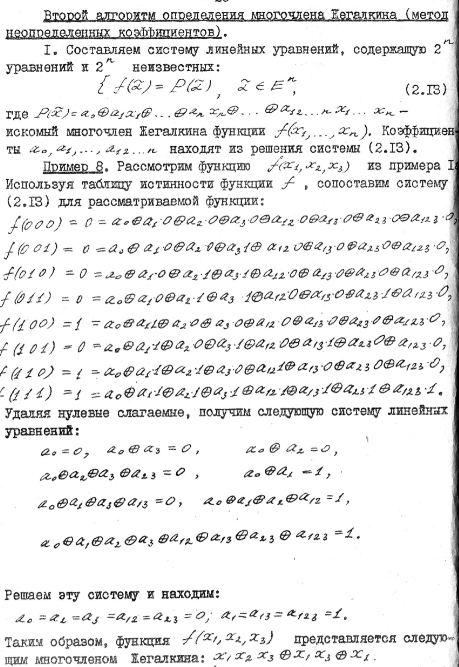
Ответ.

*Определение.*



*Для практических нужд:*

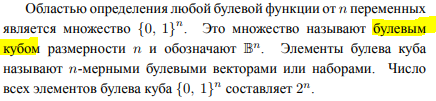
1.

2.

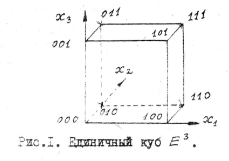
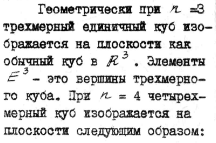
# 21.Булев куб; его грани и способы их задания.

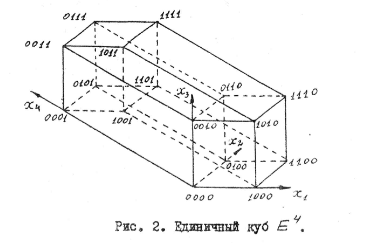
Ответ.

*Определение.*



*Примеры кубов:*





*Грани булева куба и способы их задания:*

В *кубическом представлении булевой функции от n переменных* все множество из 2n наборов ее аргументов рассматривается как *множество координат вершин n-мерного куба* с длиной ребра, равной 1. В соответствии с этим, наборы аргументов, на которых булева функция принимает значение, равное 1, принято называть существенными вершинами. *Существенные вершины* образуют так называемые ноль-кубы (0-кубы). Между *0-кубами* существует отношение соседства и определена операция склеивания. Два *0- куба* называются *соседними*, если они *отличаются только по одной координате и, соответственно, могут вступать в операцию склеивания*, в результате которой получается 1- куб.

**Пример.** Для функции f 4(X) определить, являются ли ее 0- кубы (0101) и (0001) соседними и, если являются, то выполнить операцию их склеивания.

*Решение.*

Заданные кубы являются соседними, так как они различаются только по одной координате. Результатом их склеивания является 1-куб (0Х01).

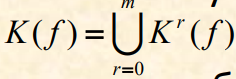
Координата, отмечаемая символом Х, называется свободной (независимой, несвязанной), а остальные (числовые), координаты называются зависимыми (связанными). Аналогичное отношение соседства существует между 1- кубами, в результате склеивания которых получается 2- куб.

**Пример.** Для функции f 4(X) определить, являются ли ее 1- кубы (0Х01) и (0Х11) соседними и, если являются, то выполнить операцию их склеивания.

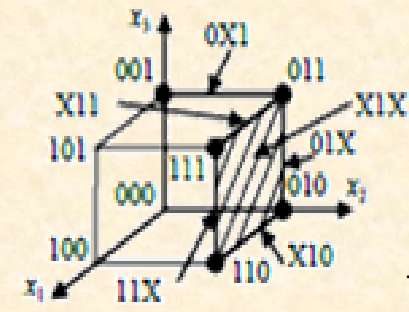
*Решение.*

Заданные кубы являются соседними, так как они различаются только по одной координате. Результатом их склеивания является 2-куб (0ХХ1).

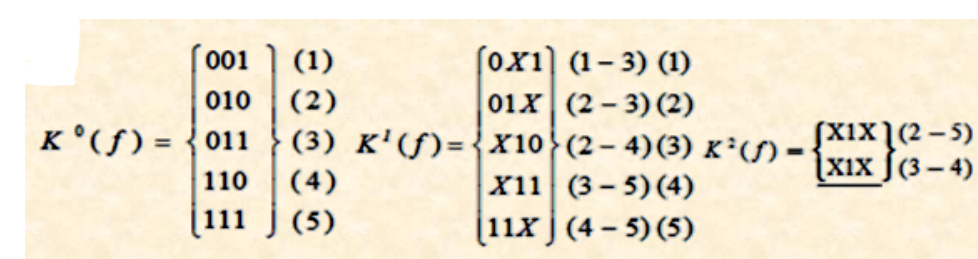
Для дальнейшего понимания, очень важны следующие термины:

**Кубическим комплексом K0(f) булевой функции f** называется множество 0-кубов этой функции. В общем случае, **кубическим комплексом K(f) булевой функции f** называется объединение множеств кубов всех размерностей этой функции , m - максимальная размерность кубов функции f.

**Пример.** Получить кубический комплекс функции .

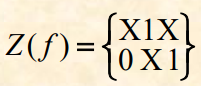
*Решение.*

Для получения кубического комплекса K(f) необходимо провести всевозможные операции склеивания над 0-кубами, 1-кубами и т.д. до тех пор, пока при склеивании r-кубов не получится Kr+1(f)= ∅.

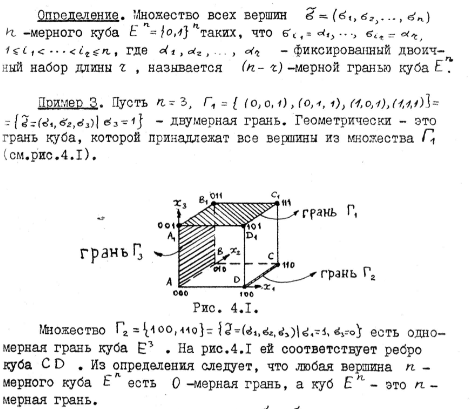


K3(f)=∅ (пустое множество). K(f)= K0(f) ˅ K1(f) ˅ K2(f). Естественно, что в комплексе K2(f) останется только один куб (Х1Х).

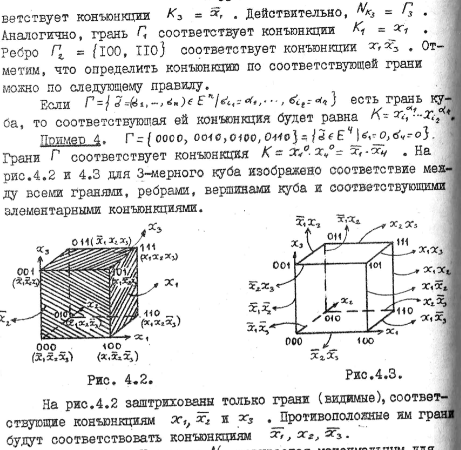
Куб, входящий в состав кубического комплекса K(f), называется ***максимальным***, если он не вступает ни в одну операцию склеивания.

Множество максимальных кубов функции f обозначается **Z(f)**. Это множество является окончательным результатом операции склеивания кубов. Из кубов этого множества (и только из них) строится ***минимальное покрытие булевой функции***. В примере максимальными кубами являются кубы .

*(Некоторое разъяснение лично от меня). n-куб и n-мерный куб — это разные понятия.*



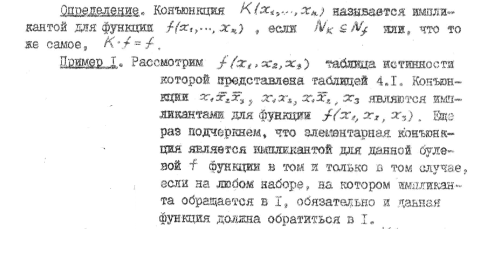
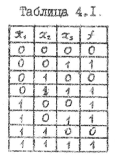


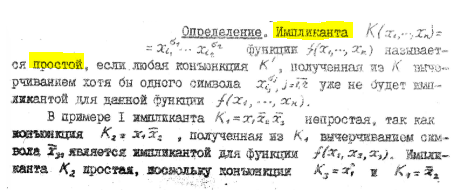


# 22.Сокращенная дизъюнктивная нормальная форма.

Ответ.

Для понятного определения **сокращённой дизъюнктивной нормальной формы**, воспользуемся несколькими терминами:

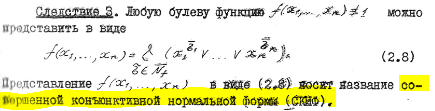


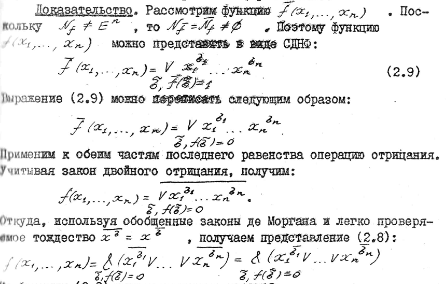


*Определение.*

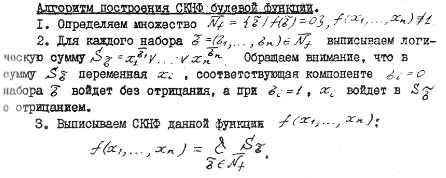


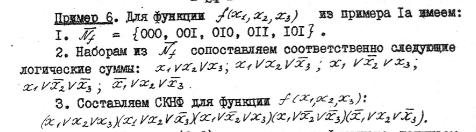
Для практических нужд необходим *алгоритм нахождения сокращённой ДНФ*, но для него нужно знать определение *Совершенной Конъюнктивной Формы (СКНФ)*, которое является *3 следствием* *Теоремы о разложении булевой функции по переменным* (эту теорему мы рассматривали в 19 вопросе):



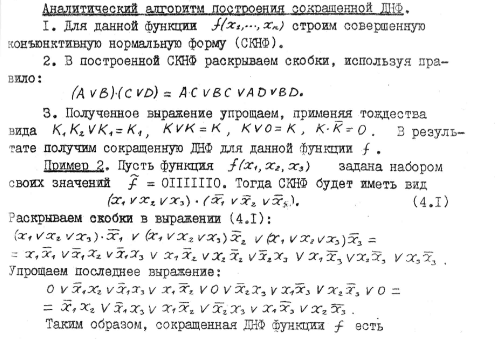


Для практических нужд нужен алгоритм нахождения СКНФ:



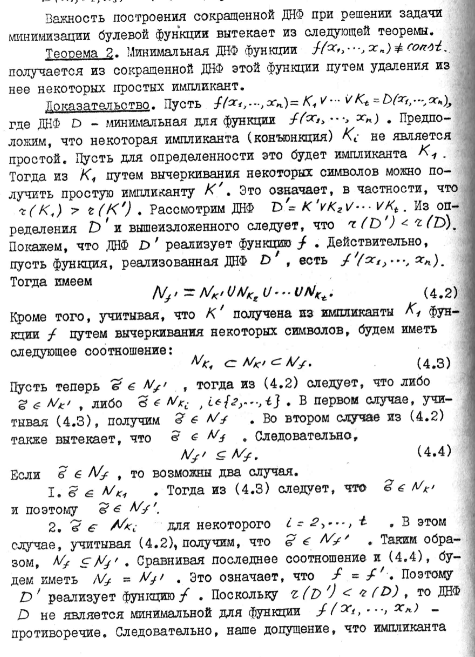


А теперь то, к чему это велось:





Прикреплю сюда блок «для тех, кому интересно зачем это»:





# 23.Задача о нахождении тупикового покрытия; тупиковое покрытие.

Ответ.

Хочется сделать очень большое лирическое отступление.

**Аналогия между импликантами и кубическим представлением булевой функции:**

*Любому кубу из К(f) можно поставить в соответствие конъюнктивный терм, который можно рассматривать как импликанту булевой функции.*

*Любой простой импликанте булевой функции соответствует максимальный куб, и, в свою очередь, множество всех простых импликант соответствует множеству Z(f) всех максимальных кубов К(f).*

Таким образом, можно провести некоторую аналогию между сокращенной СДНФ и Z(f).

В отношении импликант булевой функции так же, как и в отношении кубов, соответствующих им, существует отношение покрытия.

Принято считать, что импликанта булевой функции покрывает некоторую существенную вершину этой функции или, в общем случае, некоторый куб из К(f), если значение импликанты на наборе аргументов, представляющем данную существенную вершину, равно 1 или, в общем случае, значение импликанты равно 1 для всех существенных вершин, покрываемых кубом из К(f).

**Например,** импликанта х1х2 покрывает существенные вершины (110, 111) и в свою очередь покрывает куб 11Х.

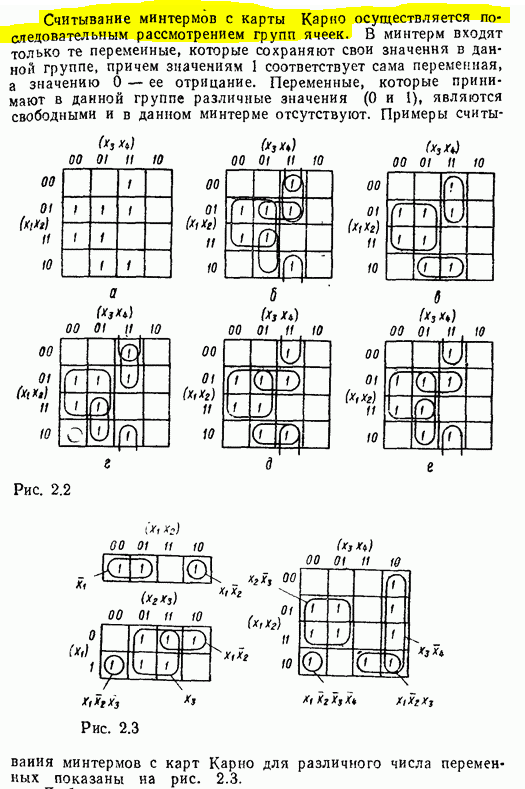
Множество импликант булевой функции образует ***полную систему импликант***, если любая существенная вершина булевой функции покрывается хотя бы одной импликантой этого множества.

Так, например, кубам из кубического комплекса К°(f) соответствует *полная система импликант*, представляющая собой множество конституент 1 данной функции f. В свою очередь, множеству максимальных кубов Z(f), естественно образующих покрытие булевой функции, соответствует полная система простых импликант.

Система простых импликант называется ***приведенной***, если она является полной, а никакая ее собственная часть уже не образует полную систему импликант.

**Карты Карно:**





Возвращаемся к теме.

Для начала поймем, что такое покрытие (советую перед разбором этого билета прочитать 21 билет):

Определение. **Покрытием булевой функции f** называется такое подмножество кубов из *кубического комплекса K(f)*, которое покрывает все *существенные вершины функции*.

Определение: **Тупиковым покрытием** называют покрытие, при удалении любого интервала из которого, оно перестает быть покрытием.

**Утверждение.**

*Множество тупиковых покрытий функции*https://studfiles.net/html/2706/21/html_1Xbfw2ZE4n.Zd9U/img-7YiTwf.png*есть в точности множество всех максимальных интервалов функции покрытия*https://studfiles.net/html/2706/21/html_1Xbfw2ZE4n.Zd9U/img-JprCip.png*.*

Легко установить следующее.

Функция покрытия есть суперпозиция монотонных функций логического сложения и логического умножения, которые монотонны. Поэтому и функция покрытия монотонна. В силу этого *допустимые интервалы функции (без отрицаний переменных) https://studfiles.net/html/2706/21/html_1Xbfw2ZE4n.Zd9U/img-Bnmz_b.pngи покрытия единицhttps://studfiles.net/html/2706/21/html_1Xbfw2ZE4n.Zd9U/img-Z9ZKr2.pngмаксимальными интервалами взаимно однозначно соответствуют друг другу (соответствие тривиальное).* При этом ***максимальным интерваламhttps://studfiles.net/html/2706/21/html_1Xbfw2ZE4n.Zd9U/img-kaqK_U.pngвзаимно однозначно соответствуют тупиковые покрытия единиц функцииhttps://studfiles.net/html/2706/21/html_1Xbfw2ZE4n.Zd9U/img-XGqIy1.png.*** Поэтому задача минимизации двоичной функции в виде ДНФ есть поиск сокращенных ДНФ двух функций - начальной и функции покрытия. Причем вторая задача применяется к монотонной функции.

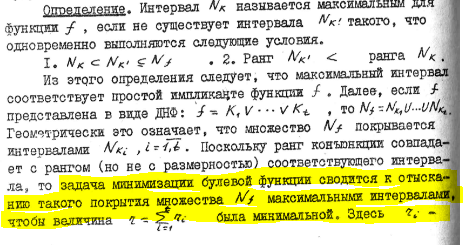
**Замечание.**

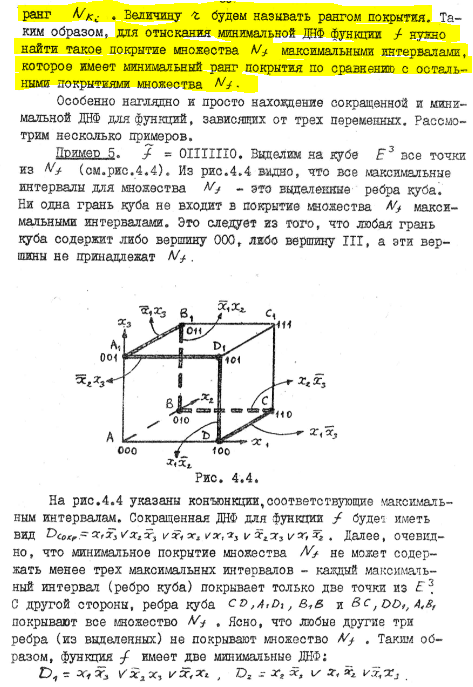
*На практике удобно применять следующее соответствие.* Максимальные интервалы монотонной функции получаются из минимальных единиц перечислением тех переменных https://studfiles.net/html/2706/21/html_1Xbfw2ZE4n.Zd9U/img-o7J1_E.png, которые равны единице в данном наборе. Например, если минимальная единица набор- 0110, то максимальный интервалhttps://studfiles.net/html/2706/21/html_1Xbfw2ZE4n.Zd9U/img-5ZTPvq.png. Повторяя эту операцию ко всем минимальным единицам функцииhttps://studfiles.net/html/2706/21/html_1Xbfw2ZE4n.Zd9U/img-jpkhcA.pngполучим все максимальные интервалыhttps://studfiles.net/html/2706/21/html_1Xbfw2ZE4n.Zd9U/img-KTziCU.png.

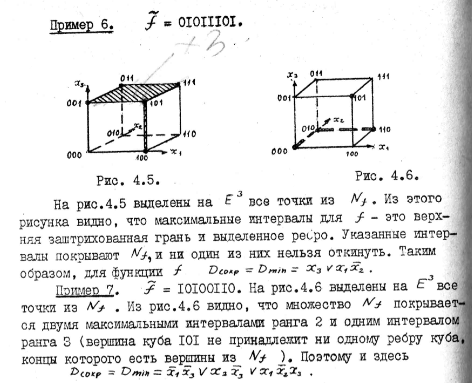
(от себя) То есть задача о нахождении тупикового покрытия равносильна задачи минимизации булевой функции. Почти всё различие заключается в подходе рассмотрения задачи. Таким образом, сейчас мы должны рассмотреть методы нахождения ***минимальной ДНФ***.

**Минимальная ДНФ** — такая *сокращенная ДНФ*, в которой содержится минимальное количество вхождений переменных.

Алгоритм нахождения сокращ енной ДНФ и метод удаления из неё некоторых простых импликант для нахождения минимальной ДНФ был рассмотрен в **22 вопросе**. Но существует ещё и геометрическая постановка задачи минимизации булевых функций. Давайте рассмотрим её, а также метод нахождения минимальной ДНФ с помощью геометрического представления.





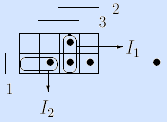


# 24.«Максимальный интервал» — максимальная (по включению) грань куба, содержащаяся в носителе булевой функции.

Ответ.

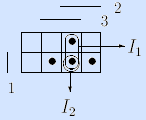
**Определение.** Интервал назовем *допустимым для булевой функции*, если на всех его наборах функция равна 1.

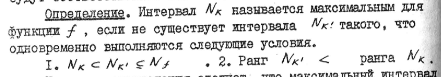
**Примеры.** I1= – 1 1 – допустимый интервал для мажоритарной функции, I2= 1 0 – – не допустимый.

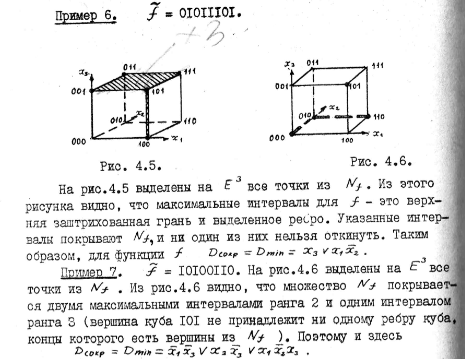


**Определение.** Интервал I назовем *максимальным для булевой функции f(x1, …, xn)*, если он является допустимым для этой функции, и не существует другого допустимого интервала I', такого что I https://ido.tsu.ru/iop_res/bulevfunc/text/img/img5.gif I'.

**Пример.** I1= –11 является максимальным интервалом для мажоритарной функции, а допустимый интервал I2 = 111 не является максимальным, так как I2https://ido.tsu.ru/iop_res/bulevfunc/text/img/img5.gif I1.





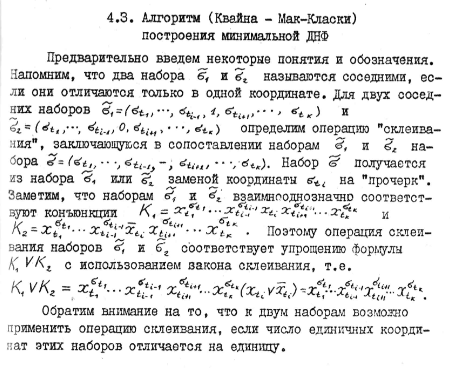


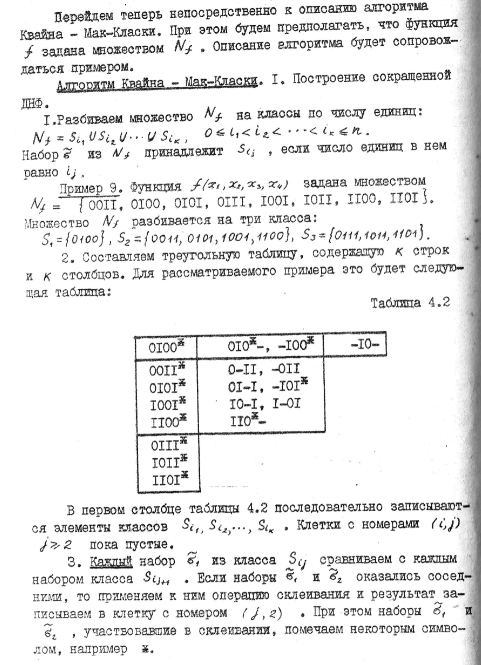
# 25.Матрица инциденции покрытия

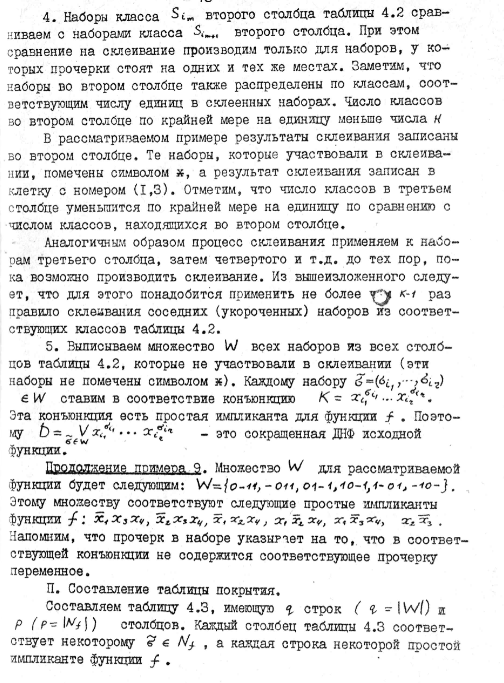
Ответ.

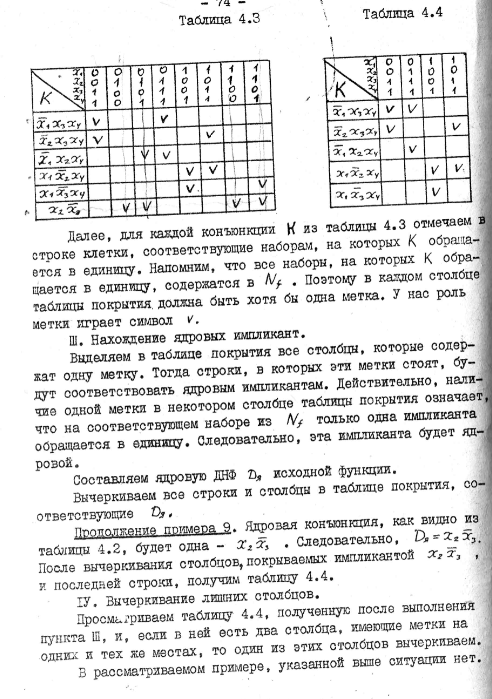
**Матрица инциденции покрытия** – матрица, которая используется в методе Квайна - Мак-Класки.

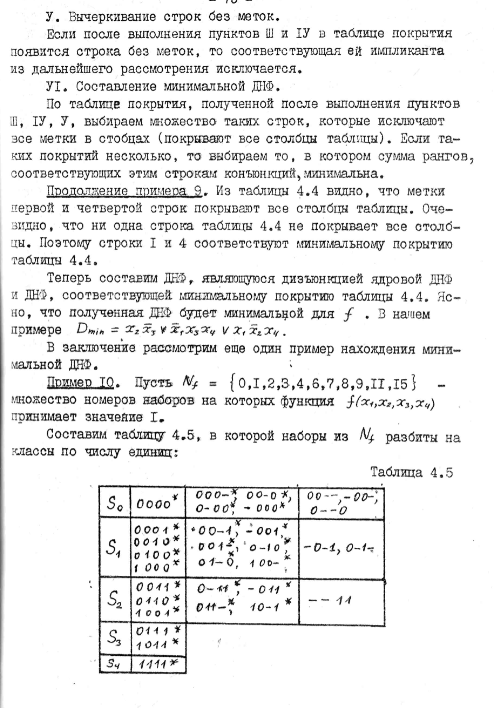
Давайте рассмотрим подробно алгоритм построения минимальной ДНФ методом Квайна – Мак-Класки и разберем подробно 2 примера:

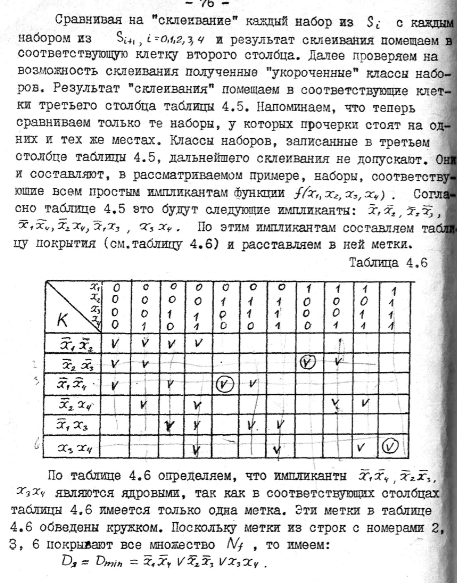






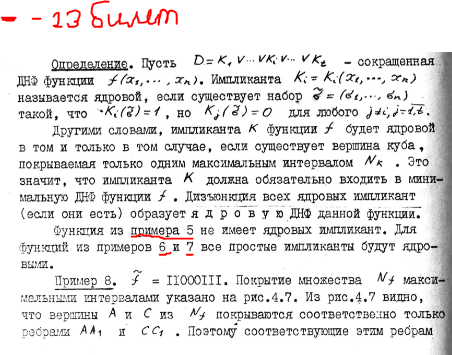


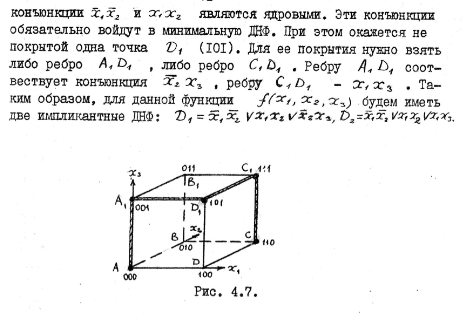




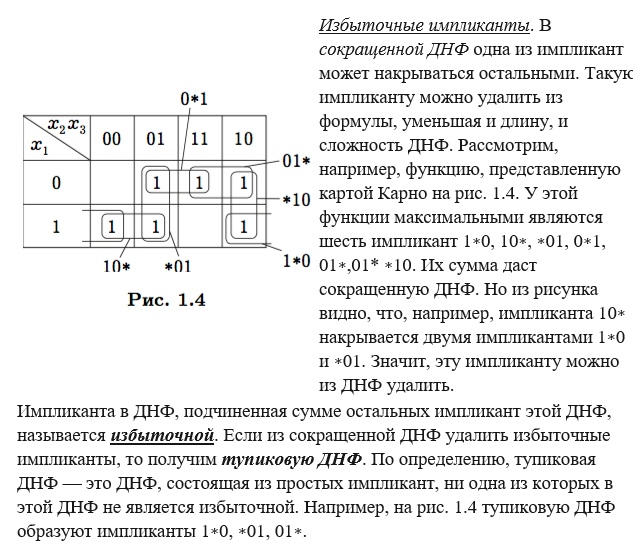
# 26.Ядро покрытия; ядровая ДНФ.

Ответ.





# 27. Задача о нахождении тупиковых ДНФ булевой функции.



# 28. Схема из функциональных элементов; реализация булевой функции схемой из функциональных элементов

На вход подаются всевозможные наборы значений переменных, а на выходе появляются соответствующие этим наборам значения функции f, представляемой формулой.

Отождествление переменных осуществляется при помощи ветвления проводников.

Чтобы осуществить подстановку одной функции в другую нужно выход логического элемента, который реализует первую функцию, направить на вход логического элемента, который реализует вторую функцию.

Каждый *Функциональный элемент* имеет n упорядоченных входов и один выход

# 29. Полная система функций

Система называется полной, если любую булеву функцию можно выразить в виде формулы через функции этой системы. По теореме Поста: если f не лежит ни в одном из 5 классов (Т0 Т1 S M L) то это полная система.

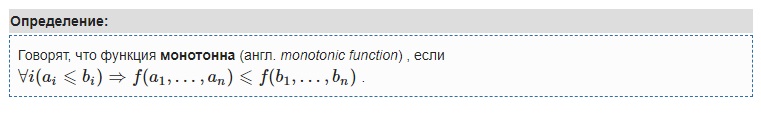
# 30. Замкнутый класс функций

Замкнутый класс в теории булевых функций — такое множество, замыкание которого относительно операции суперпозиции совпадает с ним самим: {\displaystyle [P]=P}. Другими словами, любая функция, которую можно выразить формулой с использованием функций множества {\displaystyle P}, снова входит в это же множество.

Замыканием множества функций называется такое подмножество всех булевых функций, что любую из этих функций можно выразить через функции исходного множества.

Суперпозиция функций (или сложная функция, или композиция функций) — это функция, полученная из некоторого множества функций путем подстановки одной функции в другую или отождествления переменных. Множество всех возможных не эквивалентных друг другу суперпозиций данного множества функций образует замыкание данного множества функций.

# 31. Монотонная булева функция. Критерий монотонности в терминах сокращенной ДНФ.



Булева функция f(x1, …, xn) называется монотонной (принадлежит классу M), если для любой пары наборов α и β таких, что α β, выполняется условие f(α)≤ f(β) (назовем его условием монотонности).

# 32. Линейная булева функция. Необходимое условие линейности ( равное число значений 0 и 1 )

Определение. Булева функция называется *линейной* (*принадлежит классу L*), если ее полином Жегалкина линеен.

Примеры. Мажоритарная функция не является линейной: степень ее полинома Жегалкина (xy http://ido.tsu.ru/iop_res/bulevfunc/text/img/img7.gif xz http://ido.tsu.ru/iop_res/bulevfunc/text/img/img7.gifyz) равна 2. Из элементарных булевых функций линейными являются, например, инверсия и эквивалентность. Не являются линейными, например, штрих Шеффера и стрелка Пирса. •

Утверждение о числе булевых функций класса L. Число различных линейных булевых функций, зависящих от *n* переменных, равно 2*n*+1.

*Доказательство.* Полином Жегалкина линейной функции f(x1, …, xn) имеет вид:

P=a0http://ido.tsu.ru/iop_res/bulevfunc/text/img/img7.gif a1x1http://ido.tsu.ru/iop_res/bulevfunc/text/img/img7.gif a2x2 http://ido.tsu.ru/iop_res/bulevfunc/text/img/img7.gif … http://ido.tsu.ru/iop_res/bulevfunc/text/img/img7.gif an xn,

где a0, …, an – булевы константы. Таким образом, каждый линейный полином определяется булевым вектором a0… an длины *n*+1, и наоборот, каждый булев вектор длины *n*+1 задает линейный полином Жегалкина некоторой функции *n* аргументов. Следовательно, число линейных полиномов (а значит, и число различных линейных функций *n* аргументов) равно числу булевых векторов длины *n*+1, то есть равно 2*n*+1. •

# 33. Булева функция, двойственная к данной. Построение таблицы значений двойственной функции.

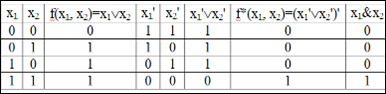
Определение. Будем называть булеву функцию f\*(x1,x2,…,xn), n1, двойственной относительно функции f(x1,x2,…,xn), если она получена из f(x1,x2,…,xn) инверсией всех аргументов и самой функции:

f\*(x1, x2,…, xn) f '(x1', x2',…, xn').

Замечание. При n=0 полагают, что функция 0 двойственна 1, а 1 двойственна 0.

Построим функции, двойственные некоторым элементарным функциям.

Пример 8. Пусть f(x1,x2)=х1х2. Покажем, что для дизъюнкции двойственной функцией будет конъюнкция. Используем таблицу истинности:



Два одинаковых правых столбца убеждают нас в справедливости доказательства.

Заметим, что то же самое получим, если применить формулы равносильных преобразований:

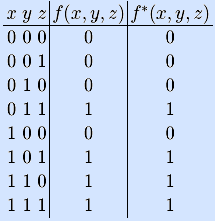
f\*(x1,x2)=(х1'х2')'=(х1&х2)''=x1&x2.

# 34. Самодвойственная булева функция. Как по таблице значений проверить самодвойственность?

Определение. Булева функция f(x1, …, xn) *самодвойственна* (*принадлежит классу S*), если она равна двойственной себе функции, то есть

f(x1, …, xn) = f\*(x1, …, xn)= f (x 1, …, x n).

Примеры. Мажоритарная функция самодвойственна:



Из элементарных функций самодвойственными являются лишь тождественная функция и инверсия. Остальные функции, в частности, штрих Шеффера и стрелка Пирса, несамодвойственны.

Алгоритм распознавания самодвойственной функции, заданной таблицей истинности. Очевидно, что для проверки самодвойственности булевой функции можно не получать двойственную ей функцию в явном виде, а лишь сравнивать значения исходной функции на противоположных наборах. Функция самодвойственна, если и только если на противоположных наборах принимает противоположые значения.

Достаточное условие несамодвойственности булевой функции. Если число единиц в столбце значений функции не совпадает с числом нулей, то функция не является самодвойственной.

# 35. Композиция булевых функций

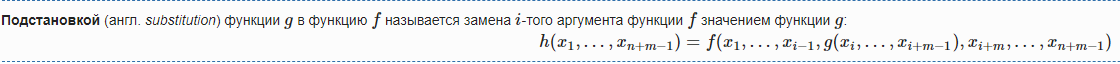
|  |
| --- |
| **Суперпозиция функций** (или **сложная функция**, или **композиция функций**) — это функция, полученная из некоторого множества функций путем подстановки одной функции в другую или отождествления переменных. |

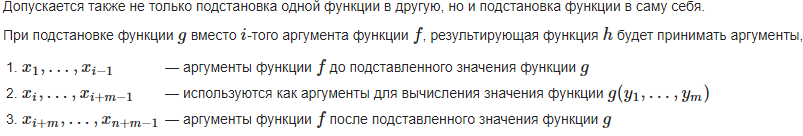
Множество всех возможных не эквивалентных друг другу суперпозиций данного множества функций образует [замыкание](https://neerc.ifmo.ru/wiki/index.php?title=%D0%9F%D1%80%D0%B5%D0%B4%D1%81%D1%82%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D0%B5_%D1%84%D1%83%D0%BD%D0%BA%D1%86%D0%B8%D0%B8_%D1%84%D0%BE%D1%80%D0%BC%D1%83%D0%BB%D0%BE%D0%B9,_%D0%BF%D0%BE%D0%BB%D0%BD%D1%8B%D0%B5_%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D1%8B_%D1%84%D1%83%D0%BD%D0%BA%D1%86%D0%B8%D0%B9) данного множества функций.

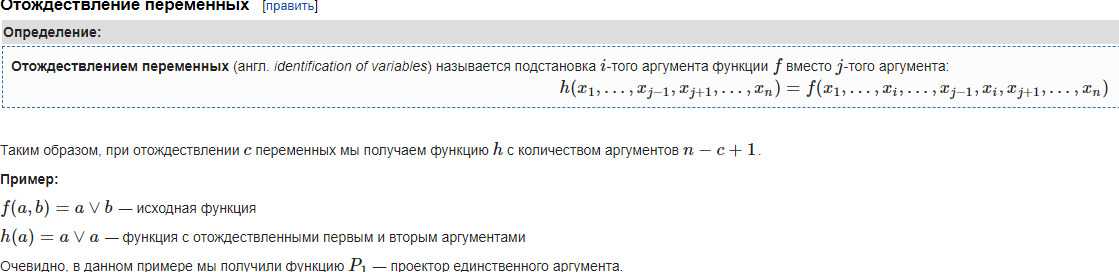
Рассмотрим две [булевы функции](https://neerc.ifmo.ru/wiki/index.php?title=%D0%9E%D0%BF%D1%80%D0%B5%D0%B4%D0%B5%D0%BB%D0%B5%D0%BD%D0%B8%D0%B5_%D0%B1%D1%83%D0%BB%D0%B5%D0%B2%D0%BE%D0%B9_%D1%84%D1%83%D0%BD%D0%BA%D1%86%D0%B8%D0%B8): функцию ff от nn аргументов f(x1,x2,…,xn)f(x1,x2,…,xn) и функцию gg от mm аргументов g(y1,y2,…,ym)g(y1,y2,…,ym).

Тогда мы можем получить новую функцию из имеющихся двумя способами:

1. Подстановкой одной функции в качестве некоторого аргумента для другой;
2. Отождествлением аргументов функций.

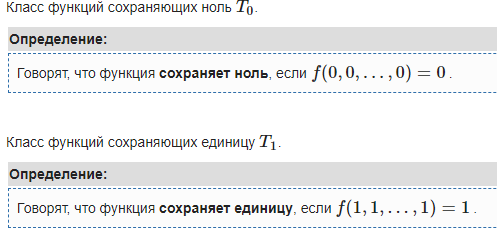


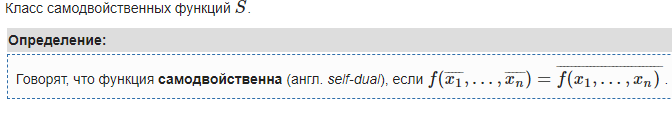




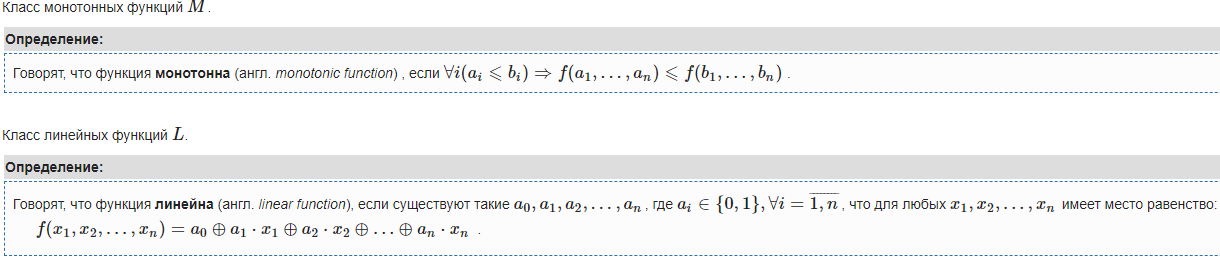
**Ранг суперпозиции** — это минимальное число подстановок и отождествлений, за которое суперпозиция может быть получена из исходного множества функций. Суперпозиция KK ранга nn обозначается как Kn.

# 36. Пять основных замкнутых классов





 Иными словами, функция называется самодвойственной, если на противоположных наборах она принимает противоположные значения.



Количество линейных функций от nn переменных равно  2^(n+1).

Функция является линейной тогда, и только тогда, когда в ее [полиноме Жегалкина](https://neerc.ifmo.ru/wiki/index.php?title=%D0%9F%D0%BE%D0%BB%D0%B8%D0%BD%D0%BE%D0%BC_%D0%96%D0%B5%D0%B3%D0%B0%D0%BB%D0%BA%D0%B8%D0%BD%D0%B0) присутствуют слагаемые, каждое из которых зависит не более чем от одной переменной.

# 37. Теорема Поста

Набор булевых функций KK является полным тогда и только тогда, когда он не содержится полностью ни в одном из классов S,M,L,T0,T1S,M,L,T0,T1, иными словами, когда в нем имеется хотя бы одна функция, не сохраняющая ноль, хотя бы одна функция, не сохраняющая один, хотя бы одна несамодвойственная функция, хотя бы одна немонотонная функция и хотя бы одна нелинейная функция.

# 38.Граф

Граф — абстрактный математический объект, представляющий собой множество вершин графа и набор рёбер, то есть соединений между парами вершин. Например, за множество вершин можно взять множество аэропортов, обслуживаемых некоторой авиакомпанией, а за множество рёбер взять регулярные рейсы этой авиакомпании между городами.

Граф - это совокупность объектов со связями между ними. Объекты рассматриваются как вершины, или узлы графа, а связи - как дуги, или ребра.

## Способы задания графа

1. графический;

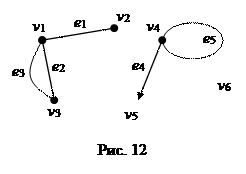
2. с помощью перечисления ребер;

3. с помощью матрицы смежности;

4. с помощью матрицы инцидентности.

1) Графический способ.

Вершины изображают точками на плоскости, а ребра – линиями, соединяющими соответствующие точки. Для изображения дуги используется линия со стрелкой, указывающей направление от начала к концу дуги.

На рисунке 12 изображен смешанный граф с вершинами v1, v2,¼, v6, ребрами e1, e2, e3, e5 и дугой e4. Смежные вершины v1, v2, инциденты ребру e1. Вершины v1, v3, инциденты параллельным ребрам e2 и e3. Вершине v4 инциденты петля e5 и дуга e4, причем v4 является началом дуги e4, а v5 – концом этой дуги. Степень вершины v1 равна 3, вершины v2 – 1, вершины v3 – 2, вершины v4 – 3, вершины v5 – 1, вершины v6 – 0. Таким образом, вершины v2 и v5 – висячие, а вершина v6 – изолированная. В случае дуги e4 точнее было бы говорить о полустепенях исхода и захода вершин v4 и v5, а именно: полустепень исхода вершины v4 равна 3, вершины v5 – 0, полустепень захода вершины v4 равна 2, вершины v5 – 1.

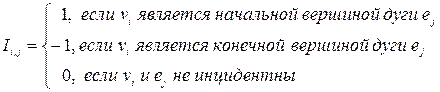
2) Аналитический способ.

Граф задают перечислением элементов множества вершин и множества ребер. Для графа, изображенного на рисунке 12, эти множества: V={v1, v2, v3, v4, v5, v6} и Е={e1, e2, e3, e4, e5}, где e1=(v1, v2), e2=(v1, v3), e3=(v1, v3), e4=(v4, v5), и e5=(v4, v4).

3) Матричный способ.

Имеется несколько вариантов задать граф матрицей. Наиболее употребимыми являются матрица инциденций и матрица смежности.

а) Матрица инциденций – это прямоугольная матрица, число строк которой равно числу вершин, а число столбцов – числу дуг (ребер) графа. Элементы этой матрицы определяются следующим образом:



Таким образом, для графа на рисунке 12 матрица инциденций такова:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  | e1 | e2 | e3 | e4 | e5 |
|  | v1 |  |  |  |  |  |
|  | v2 |  |  |  |  |  |
| I= | v3 |  |  |  |  |  |
|  | v4 |  |  |  |  |  |
|  | v5 |  |  |  | -1 |  |
|  | v6 |  |  |  |  |  |

По этой матрице легко судить о наличии в графе параллельных ребер (два одинаковых столбца), петли (одна единица в столбце), дуги (значения разных знаков в столбце), изолированной вершины (нулевая строка), висячих вершин (одно ненулевое значение в строке).

б) Матрица смежности вершин – это квадратная матрица, размер которой определяется числом вершин в графе. Элементы этой матрицы определяются так: http://helpiks.org/helpiksorg/baza5/114510670208.files/image005.gif. Если в графе имеются параллельные ребра, то соответствующий элемент матрицы смежности полагают равным числу этих ребер. Так матрица смежности для графа на рисунке 12 такова:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | v1 | v2 | v3 | v4 | v5 | v6 |
|  | v1 |  |  |  |  |  |  |
|  | v2 |  |  |  |  |  |  |
| S= | v3 |  |  |  |  |  |  |
|  | v4 |  |  |  |  |  |  |
|  | v5 |  |  |  |  |  |  |
|  | v6 |  |  |  |  |  |  |

По виду этой матрицы также несложно судить о наличии в графе кратных ребер, дуг, петель, висячих и изолированных вершин.

## Матрица смежности

Матрицей смежности A=||αi,j|| невзвешенного графа G=(V,E) называется матрица A[V×V], в которой αi,j — количество рёбер, соединяющих вершины vi и vj, причём при i=j каждую петлю учитываем дважды, если граф не является ориентированным, и один раз, если граф ориентирован.

Матрицей смежности A=||αi,j|| взвешенного графа G=(V,E) называется матрица A[V×V], в которой αi,j — вес ребра, соединяющего вершины vi и vj.

Утверждения

Для графов без петель и кратных рёбер матрица смежности бинарна (состоит из нулей и единиц).

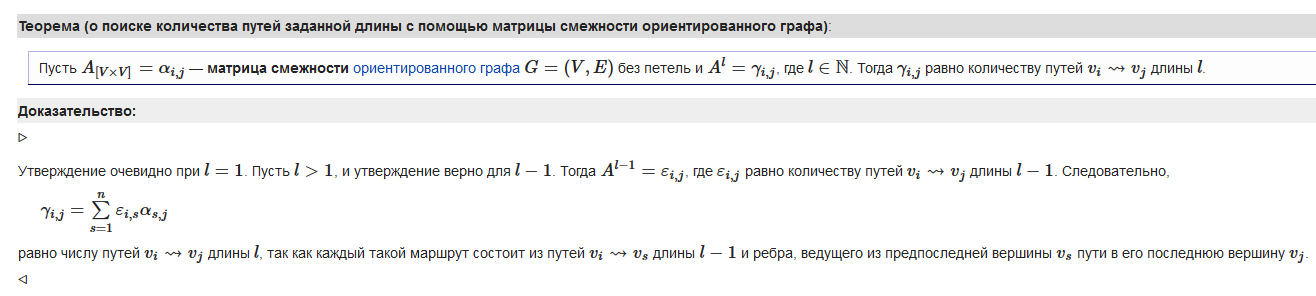
Для графов без петель и кратных рёбер главная диагональ матрицы смежности целиком состоит из нулей.



Матрица смежности является симметричной.

Д-во





## Матрица инцидентности

Матрицей инцидентности неориентированного графа называется матрица I(|V|×|E|), для которой Ii,j=1, если вершина vi инцидентна ребру ej, в противном случае Ii,j=0.

Матрицей инцидентности ориентированного графа называется матрица I(|V|×|E|), для которой Ii,j=1, если вершина vi является началом дуги ej, Ii,j=−1, если vi является концом дуги ej, в остальных случаях Ii,j=0.

Утверждения

Для неориентированных графов без петель и кратных рёбер матрица инцидентности бинарна (состоит из нулей и единиц).

Для ориентированных графов без петель и кратных рёбер матрица инцидентности состоит из нулей, единиц и −1.

Сумма элементов i-й строки равна degvi.

Сумма элементов i-й строки равна deg+vi−deg-vi.

# 39.Автоморфизм [графа](https://ru.wikipedia.org/wiki/%D0%93%D1%80%D0%B0%D1%84_(%D1%82%D0%B5%D0%BE%D1%80%D0%B8%D1%8F_%D0%B3%D1%80%D0%B0%D1%84%D0%BE%D0%B2))

Автоморфизм [графа](https://ru.wikipedia.org/wiki/%D0%93%D1%80%D0%B0%D1%84_(%D1%82%D0%B5%D0%BE%D1%80%D0%B8%D1%8F_%D0%B3%D1%80%D0%B0%D1%84%D0%BE%D0%B2)) есть отображение множества вершин на себя, сохраняющее смежность. Множество таких автоморфизмов образует вершинную группу графа или просто группу графа. Группа подстановок на множестве ребер называется реберной группой графа, которая тесно связана с вершинной:

Реберная и вершинная группы графа изоморфны тогда и только тогда, когда имеется не более одной [изолированной вершины](https://ru.wikipedia.org/wiki/%D0%A1%D0%BB%D0%BE%D0%B2%D0%B0%D1%80%D1%8C_%D1%82%D0%B5%D1%80%D0%BC%D0%B8%D0%BD%D0%BE%D0%B2_%D1%82%D0%B5%D0%BE%D1%80%D0%B8%D0%B8_%D0%B3%D1%80%D0%B0%D1%84%D0%BE%D0%B2#И), и нет [компонент связности](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BC%D0%BF%D0%BE%D0%BD%D0%B5%D0%BD%D1%82%D0%B0_%D1%81%D0%B2%D1%8F%D0%B7%D0%BD%D0%BE%D1%81%D1%82%D0%B8_%D0%B3%D1%80%D0%B0%D1%84%D0%B0) состоящих из единственного ребра.

Граф, для которого единственный возможный автоморфизм это тождественное отображение, называется асимметрическим. Наименьшее асимметрическое дерево имеет семь вершин, а наименьший асимметрический граф шесть вершин и столько же ребер.

Для любой конечной группы найдется такой конечный неориентированный граф, что его группа автоморфизмов изоморфна данной. Результат получен Р. Фрухтом, в основе доказательства — преобразование цветного графа группы, обобщения [графа Кэли](https://ru.wikipedia.org/wiki/%D0%93%D1%80%D0%B0%D1%84_%D0%9A%D1%8D%D0%BB%D0%B8).

# 40.Изоморфизм графов

Определение. Два графа G1 и G2 называются изоморфными, если существует взаимно однозначное соответствие между множествами их вершин, обладающее тем свойством, что число ребер, соединяющих любые две вершины в G1, равно число ребер, соединяющих соответствующие вершины в G2.

Из определения следует, что изоморфные графы можно одинаково изображать графически и отличаться они будут только метками вершин.

Изоморфизм графов есть отношение эквивалентности.

## Гомеоморфизм графов

Орграф – ориентированный граф

Определение. Операция подразбиения (измельчения) дуги (u, v) в орграфе D = (V, E) состоит в удалении из Е дуги (u, v), добавлении к V новой вершины w и добавлении к Е | {(u, v)} двух дуг  (u, v), (w, v). Аналогично определяется операция подразбиения ребра в графах.

Определение. Орграф D1 называется подразбиением орграфа D2, если орграф D1 можно получить из D2 путем последовательного применения операции подразбиения дуг. Аналогично определяется подразбиение графа.

Определение. Орграфы D1, D2 называются гомеоморфными, если существуют их подразбиения, являющиеся изоморфными.

## Гомотопическая эквивалентность графов

Гомотопия

Два непрерывных отображения топологического пространства http://dict.sernam.ru/htm/0/EX7ta3edTr/5.files/image001.gif в топологическое пространство http://dict.sernam.ru/htm/0/EX7ta3edTr/5.files/image002.gif

http://dict.sernam.ru/htm/0/EX7ta3edTr/5.files/image003.gif и http://dict.sernam.ru/htm/0/EX7ta3edTr/5.files/image004.gif,

называются гомотопными, если существует семейство отображений

http://dict.sernam.ru/htm/0/EX7ta3edTr/5.files/image005.gif, http://dict.sernam.ru/htm/0/EX7ta3edTr/5.files/image006.gif,

удовлетворяющее таким свойствам:

1. http://dict.sernam.ru/htm/0/EX7ta3edTr/5.files/image007.gif непрерывна по совокупности аргументов http://dict.sernam.ru/htm/0/EX7ta3edTr/5.files/image008.gif и http://dict.sernam.ru/htm/0/EX7ta3edTr/5.files/image009.gif,

2. http://dict.sernam.ru/htm/0/EX7ta3edTr/5.files/image010.gif, http://dict.sernam.ru/htm/0/EX7ta3edTr/5.files/image011.gif.

Примеры

1. Вложение окружности в двумерную сферу в качестве экватора гомотопно отображению окружности в любую точку этой сферы.

2. Вложения окружности в тор в качестве меридиана и параллели не гомотопны между собой.

Гомотопическая эквивалентность.

Два топологических пространства http://dict.sernam.ru/htm/0/5PKhhzaR9K/2.files/image001.gif и http://dict.sernam.ru/htm/0/5PKhhzaR9K/2.files/image002.gif называются гомотопическими эквивалентами, если существуют два непрерывных отображения

http://dict.sernam.ru/htm/0/5PKhhzaR9K/2.files/image003.gif и http://dict.sernam.ru/htm/0/5PKhhzaR9K/2.files/image004.gif,

удовлетворяющие следующему свойству: отображения

http://dict.sernam.ru/htm/0/5PKhhzaR9K/2.files/image005.gif и http://dict.sernam.ru/htm/0/5PKhhzaR9K/2.files/image006.gif

(каждое в отдельности) гомотопны соответственно тождественным отображениям http://dict.sernam.ru/htm/0/5PKhhzaR9K/2.files/image007.gif, http://dict.sernam.ru/htm/0/5PKhhzaR9K/2.files/image008.gif.

Примеры

1. Двумерная плоскость гомотопически эквивалентна точке.

2. Окружность комплексной плоскости http://dict.sernam.ru/htm/0/5PKhhzaR9K/2.files/image009.gif гомотопически эквивалентна кольцу http://dict.sernam.ru/htm/0/5PKhhzaR9K/2.files/image010.gif.

3. Окружность и двумерная сфера не являются гомотопически эквивалентными.

# 41.Цикломатическое число

Цикломатическоечисло — термин теории графов, одна из возможных числовых характеристик несвязного графа. Если [граф](http://economic_mathematics.academic.ru/1241/%D0%93%D1%80%D0%B0%D1%84) X имеет n вершин, m ребер, а p — количество его  связных частей — компонент (см. Граф), то Ц.ч. определяется равенством

v(X) =  m — n + p.

При Ц.ч., равном нулю, [граф](http://economic_mathematics.academic.ru/1241/%D0%93%D1%80%D0%B0%D1%84) не содержит циклов, если же оно равно единице, то граф имеет только один [цикл](http://economic_mathematics.academic.ru/4978/%D0%A6%D0%B8%D0%BA%D0%BB).плана

# 42.Путь в графе

Путь в [графе](https://ru.wikipedia.org/wiki/%D0%93%D1%80%D0%B0%D1%84_(%D0%BC%D0%B0%D1%82%D0%B5%D0%BC%D0%B0%D1%82%D0%B8%D0%BA%D0%B0)) — последовательность [рёбер](https://ru.wikipedia.org/wiki/%D0%93%D0%BB%D0%BE%D1%81%D1%81%D0%B0%D1%80%D0%B8%D0%B9_%D1%82%D0%B5%D0%BE%D1%80%D0%B8%D0%B8_%D0%B3%D1%80%D0%B0%D1%84%D0%BE%D0%B2#ребро) (в неориентированном графе) и/или дуг (в ориентированном графе), такая, что конец одной дуги (ребра) является началом другой дуги (ребра). Или последовательность вершин и дуг (рёбер), в которой каждый элемент инцидентен предыдущему и последующему. Может рассматриваться как частный случай [маршрута](https://ru.wikipedia.org/wiki/%D0%93%D0%BB%D0%BE%D1%81%D1%81%D0%B0%D1%80%D0%B8%D0%B9_%D1%82%D0%B5%D0%BE%D1%80%D0%B8%D0%B8_%D0%B3%D1%80%D0%B0%D1%84%D0%BE%D0%B2#маршрут).

## Простой путь

Простой путь — путь, все рёбра которого попарно различны.

## Простой замкнутый путь???

Контур в ориентированном графе? Простой цикл?

# 43.Поиск кратчайшего пути

## Алгоритм Дейкстры

Алгори́тм Де́йкстры — [алгоритм](https://ru.wikipedia.org/wiki/%D0%90%D0%BB%D0%B3%D0%BE%D1%80%D0%B8%D1%82%D0%BC) на [графах](https://ru.wikipedia.org/wiki/%D0%93%D1%80%D0%B0%D1%84_(%D0%BC%D0%B0%D1%82%D0%B5%D0%BC%D0%B0%D1%82%D0%B8%D0%BA%D0%B0)). Находит кратчайшие пути от одной из вершин графа до всех остальных. Алгоритм работает только для графов без [рёбер](https://ru.wikipedia.org/wiki/%D0%A0%D0%B5%D0%B1%D1%80%D0%BE_(%D1%82%D0%B5%D0%BE%D1%80%D0%B8%D1%8F_%D0%B3%D1%80%D0%B0%D1%84%D0%BE%D0%B2)) отрицательного [веса](https://ru.wikipedia.org/wiki/%D0%A1%D0%BB%D0%BE%D0%B2%D0%B0%D1%80%D1%8C_%D1%82%D0%B5%D1%80%D0%BC%D0%B8%D0%BD%D0%BE%D0%B2_%D1%82%D0%B5%D0%BE%D1%80%D0%B8%D0%B8_%D0%B3%D1%80%D0%B0%D1%84%D0%BE%D0%B2#В).

Дан [взвешенный](https://ru.wikipedia.org/wiki/%D0%A1%D0%BB%D0%BE%D0%B2%D0%B0%D1%80%D1%8C_%D1%82%D0%B5%D1%80%D0%BC%D0%B8%D0%BD%D0%BE%D0%B2_%D1%82%D0%B5%D0%BE%D1%80%D0%B8%D0%B8_%D0%B3%D1%80%D0%B0%D1%84%D0%BE%D0%B2#В) [ориентированный](https://ru.wikipedia.org/wiki/%D0%9E%D1%80%D0%B8%D0%B5%D0%BD%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%BD%D1%8B%D0%B9_%D0%B3%D1%80%D0%B0%D1%84)граф [G ( V , E ) {\displaystyle G(V,E)}](https://ru.wikipedia.org/wiki/%D0%90%D0%BB%D0%B3%D0%BE%D1%80%D0%B8%D1%82%D0%BC_%D0%94%D0%B5%D0%B9%D0%BA%D1%81%D1%82%D1%80%D1%8B#Обозначения) без дуг отрицательного веса. Найти кратчайшие пути от некоторой вершины a {\displaystyle a} графа G {\displaystyle G} до всех остальных вершин этого графа.

Каждой вершине из V сопоставим метку — минимальное известное расстояние от этой вершины до a. Алгоритм работает пошагово — на каждом шаге он «посещает» одну вершину и пытается уменьшать метки. Работа алгоритма завершается, когда все вершины посещены.

### Инициализация

Метка самой вершины a полагается равной 0, метки остальных вершин — бесконечности. Это отражает то, что расстояния от a до других вершин пока неизвестны. Все вершины графа помечаются как непосещённые.

### Шаг алгоритма

Если все вершины посещены, алгоритм завершается. В противном случае, из ещё не посещённых вершин выбирается вершина u, имеющая минимальную метку. Мы рассматриваем всевозможные маршруты, в которых u является предпоследним пунктом. Вершины, в которые ведут рёбра из u, назовём соседями этой вершины. Для каждого соседа вершины u, кроме отмеченных как посещённые, рассмотрим новую длину пути, равную сумме значений текущей метки u и длины ребра, соединяющего u с этим соседом. Если полученное значение длины меньше значения метки соседа, заменим значение метки полученным значением длины. Рассмотрев всех соседей, пометим вершину u как посещённую и повторим [шаг алгоритма](https://ru.wikipedia.org/wiki/%D0%90%D0%BB%D0%B3%D0%BE%D1%80%D0%B8%D1%82%D0%BC_%D0%94%D0%B5%D0%B9%D0%BA%D1%81%D1%82%D1%80%D1%8B#Шаг).

## Поиск в ширину

Поиск в ширину — метод [обхода графа](https://ru.wikipedia.org/w/index.php?title=%D0%9E%D0%B1%D1%85%D0%BE%D0%B4_%D0%B3%D1%80%D0%B0%D1%84%D0%B0&action=edit&redlink=1) и [поиска пути в графе](https://ru.wikipedia.org/w/index.php?title=%D0%90%D0%BB%D0%B3%D0%BE%D1%80%D0%B8%D1%82%D0%BC_%D0%BF%D0%BE%D0%B8%D1%81%D0%BA%D0%B0&action=edit&redlink=1).

Поиск в ширину работает путём последовательного просмотра отдельных уровней [графа](https://ru.wikipedia.org/wiki/%D0%93%D1%80%D0%B0%D1%84_(%D0%BC%D0%B0%D1%82%D0%B5%D0%BC%D0%B0%D1%82%D0%B8%D0%BA%D0%B0)), начиная с узла-источника *u*u {\displaystyle u} .

Рассмотрим все [рёбра](https://ru.wikipedia.org/wiki/%D0%A0%D0%B5%D0%B1%D1%80%D0%BE_(%D1%82%D0%B5%D0%BE%D1%80%D0%B8%D1%8F_%D0%B3%D1%80%D0%B0%D1%84%D0%BE%D0%B2)) ( u , v ) {\displaystyle (u,v)} (*u,v*), выходящие из [узла](https://ru.wikipedia.org/wiki/%D0%A3%D0%B7%D0%B5%D0%BB_(%D1%82%D0%B5%D0%BE%D1%80%D0%B8%D1%8F_%D0%B3%D1%80%D0%B0%D1%84%D0%BE%D0%B2)) *u*u {\displaystyle u}uu. Если очередной узел *v {\displaystyle v} v* является целевым узлом, то поиск завершается; в противном случае узел *v {\displaystyle v} v* добавляется в [очередь](https://ru.wikipedia.org/wiki/%D0%9E%D1%87%D0%B5%D1%80%D0%B5%D0%B4%D1%8C_(%D1%81%D1%82%D1%80%D1%83%D0%BA%D1%82%D1%83%D1%80%D0%B0_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85)). поток

### Алгоритм по шагам

1. Поместить узел, с которого начинается поиск, в изначально пустую очередь.
2. Извлечь из начала очереди узел u {\displaystyle u} и пометить его как развёрнутый.
   * Если узел u {\displaystyle u} является целевым узлом, то завершить поиск с результатом «успех».
   * В противном случае, в конец очереди добавляются все преемники узла u {\displaystyle u} , которые ещё не развёрнуты и не находятся в очереди.
3. Если очередь пуста, то все узлы [связного графа](https://ru.wikipedia.org/wiki/%D0%A1%D0%B2%D1%8F%D0%B7%D0%BD%D1%8B%D0%B9_%D0%B3%D1%80%D0%B0%D1%84) были просмотрены, следовательно, целевой узел недостижим из начального; завершить поиск с результатом «неудача».
4. Вернуться к п. 2.

Примечание: деление вершин на развёрнутые и не развёрнутые необходимо для произвольного графа (так как в нём могут быть циклы). Для [дерева](https://ru.wikipedia.org/wiki/%D0%94%D0%B5%D1%80%D0%B5%D0%B2%D0%BE_(%D1%82%D0%B5%D0%BE%D1%80%D0%B8%D1%8F_%D0%B3%D1%80%D0%B0%D1%84%D0%BE%D0%B2)) эта операция не нужна, так как каждая вершина будет выбрана один-единственный раз.

# 44.Связность графа

Связный граф — граф, содержащий ровно одну компоненту связности. Это означает, что между любой парой вершин этого графа существует как минимум один путь.

## Компонента связности

Компонента связности графа — некоторое множество вершин [графа](https://dic.academic.ru/dic.nsf/ruwiki/8789) такое, что для любых двух вершин из этого множества существует путь из одной в другую, и не существует пути из вершины этого множества в вершину не из этого множества.

Для ориентированных графов определено понятие [сильной компоненты связности](https://dic.academic.ru/dic.nsf/ruwiki/647707)

Для поиска компонент связности можно использовать [поиск в ширину](https://dic.academic.ru/dic.nsf/ruwiki/130131) или [поиск в глубину](https://dic.academic.ru/dic.nsf/ruwiki/96862).

При этом затраченное время будет линейным (относительно количества вершин и ребер).

## Алгоритм проверки связности графа

## Обход в глубину

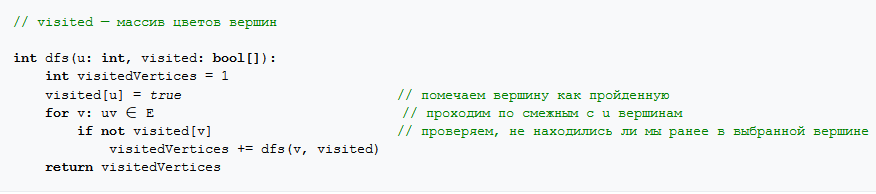
Обход в глубину— один из основных методов обхода [графа](https://neerc.ifmo.ru/wiki/index.php?title=%D0%9E%D1%81%D0%BD%D0%BE%D0%B2%D0%BD%D1%8B%D0%B5_%D0%BE%D0%BF%D1%80%D0%B5%D0%B4%D0%B5%D0%BB%D0%B5%D0%BD%D0%B8%D1%8F_%D1%82%D0%B5%D0%BE%D1%80%D0%B8%D0%B8_%D0%B3%D1%80%D0%B0%D1%84%D0%BE%D0%B2), часто используемый для [проверки связности](https://neerc.ifmo.ru/wiki/index.php?title=%D0%98%D1%81%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5_%D0%BE%D0%B1%D1%85%D0%BE%D0%B4%D0%B0_%D0%B2_%D0%B3%D0%BB%D1%83%D0%B1%D0%B8%D0%BD%D1%83_%D0%B4%D0%BB%D1%8F_%D0%BF%D1%80%D0%BE%D0%B2%D0%B5%D1%80%D0%BA%D0%B8_%D1%81%D0%B2%D1%8F%D0%B7%D0%BD%D0%BE%D1%81%D1%82%D0%B8), поиска [цикла](https://neerc.ifmo.ru/wiki/index.php?title=%D0%98%D1%81%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5_%D0%BE%D0%B1%D1%85%D0%BE%D0%B4%D0%B0_%D0%B2_%D0%B3%D0%BB%D1%83%D0%B1%D0%B8%D0%BD%D1%83_%D0%B4%D0%BB%D1%8F_%D0%BF%D0%BE%D0%B8%D1%81%D0%BA%D0%B0_%D1%86%D0%B8%D0%BA%D0%BB%D0%B0_%D0%B2_%D0%BE%D1%80%D0%B8%D0%B5%D0%BD%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%BD%D0%BE%D0%BC_%D0%B3%D1%80%D0%B0%D1%84%D0%B5) и [компонент сильной связности](https://neerc.ifmo.ru/wiki/index.php?title=%D0%98%D1%81%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5_%D0%BE%D0%B1%D1%85%D0%BE%D0%B4%D0%B0_%D0%B2_%D0%B3%D0%BB%D1%83%D0%B1%D0%B8%D0%BD%D1%83_%D0%B4%D0%BB%D1%8F_%D0%BF%D0%BE%D0%B8%D1%81%D0%BA%D0%B0_%D0%BA%D0%BE%D0%BC%D0%BF%D0%BE%D0%BD%D0%B5%D0%BD%D1%82_%D1%81%D0%B8%D0%BB%D1%8C%D0%BD%D0%BE%D0%B9_%D1%81%D0%B2%D1%8F%D0%B7%D0%BD%D0%BE%D1%81%D1%82%D0%B8) и для [топологической сортировки](https://neerc.ifmo.ru/wiki/index.php?title=%D0%98%D1%81%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5_%D0%BE%D0%B1%D1%85%D0%BE%D0%B4%D0%B0_%D0%B2_%D0%B3%D0%BB%D1%83%D0%B1%D0%B8%D0%BD%D1%83_%D0%B4%D0%BB%D1%8F_%D1%82%D0%BE%D0%BF%D0%BE%D0%BB%D0%BE%D0%B3%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%BE%D0%B9_%D1%81%D0%BE%D1%80%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%BA%D0%B8).

Общая идея алгоритма состоит в следующем: для каждой не пройденной [вершины](https://neerc.ifmo.ru/wiki/index.php?title=%D0%9E%D1%81%D0%BD%D0%BE%D0%B2%D0%BD%D1%8B%D0%B5_%D0%BE%D0%BF%D1%80%D0%B5%D0%B4%D0%B5%D0%BB%D0%B5%D0%BD%D0%B8%D1%8F_%D1%82%D0%B5%D0%BE%D1%80%D0%B8%D0%B8_%D0%B3%D1%80%D0%B0%D1%84%D0%BE%D0%B2) необходимо найти все не пройденные [смежные вершины](https://neerc.ifmo.ru/wiki/index.php?title=%D0%9E%D1%81%D0%BD%D0%BE%D0%B2%D0%BD%D1%8B%D0%B5_%D0%BE%D0%BF%D1%80%D0%B5%D0%B4%D0%B5%D0%BB%D0%B5%D0%BD%D0%B8%D1%8F_%D1%82%D0%B5%D0%BE%D1%80%D0%B8%D0%B8_%D0%B3%D1%80%D0%B0%D1%84%D0%BE%D0%B2) и повторить поиск для них.

## Алгоритм проверки связности

Дан [неориентированный граф](https://neerc.ifmo.ru/wiki/index.php?title=%D0%9E%D1%81%D0%BD%D0%BE%D0%B2%D0%BD%D1%8B%D0%B5_%D0%BE%D0%BF%D1%80%D0%B5%D0%B4%D0%B5%D0%BB%D0%B5%D0%BD%D0%B8%D1%8F_%D1%82%D0%B5%D0%BE%D1%80%D0%B8%D0%B8_%D0%B3%D1%80%D0%B0%D1%84%D0%BE%D0%B2) G=(V,E). Необходимо проверить, является ли он связным.

Снова небольшая модификация алгоритма [обхода в глубину](https://neerc.ifmo.ru/wiki/index.php?title=%D0%9E%D0%B1%D1%85%D0%BE%D0%B4_%D0%B2_%D0%B3%D0%BB%D1%83%D0%B1%D0%B8%D0%BD%D1%83,_%D1%86%D0%B2%D0%B5%D1%82%D0%B0_%D0%B2%D0%B5%D1%80%D1%88%D0%B8%D0%BD), в которой будем возвращать количество посещенных вершин. Запустим такой dfs() от некоторой вершины графа G, если его результат равен |V|, то мы побывали во всех вершинах графа, а следовательно он связен, иначе какие-то вершины остались непосещенными. Работает алгоритм за O(|V|+|E|).



# 45.Критерий эйлеровости связного графа

Граф является эйлеровым, если он содержит эйлеров цикл.

## Теорема 1

Связный граф является эйлеровым тогда и только тогда, когда каждая вершина имеет четную локальную степень.

## Теорема 2

Связный граф содержит эйлерову цепь тогда и только тогда, когда ровно две вершины имеют нечетную локальную степень.

## Теорема 3

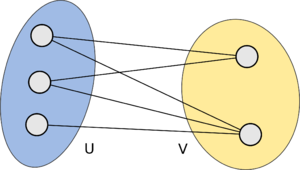
Пусть G – эйлеров граф; тогда следующая процедура всегда возможна и приводит к эйлеровой цепи графа G. Выходя из произвольной вершины и, идем по ребрам графа произвольным образом, соблюдая лишь следующие правила:  
1. стираем ребра по мере их прохождения и стираем также изолированные вершины, которые при этом образуются;  
2. на каждом этапе идем по мосту только тогда, когда нет других возможностей.

## Утверждения

1. Критерий эйлеровости: Для того, чтобы граф являлся эйлеровым необходимо и достаточно, чтобы он был связным и все его вершины имели четную степень.
2. Любой простой полный граф с нечетным количеством вершин является эйлеровым.
3. Любой циклический граф является эйлеровым. Граф, являющийся колесом, не является эйлеровым.

# **46.**Двудольные графы

Двудольный граф или биграф — [граф](https://neerc.ifmo.ru/wiki/index.php?title=%D0%9E%D1%81%D0%BD%D0%BE%D0%B2%D0%BD%D1%8B%D0%B5_%D0%BE%D0%BF%D1%80%D0%B5%D0%B4%D0%B5%D0%BB%D0%B5%D0%BD%D0%B8%D1%8F_%D1%82%D0%B5%D0%BE%D1%80%D0%B8%D0%B8_%D0%B3%D1%80%D0%B0%D1%84%D0%BE%D0%B2), множество вершин которого можно разбить на две части таким образом, что каждое ребро графа соединяет какую-то вершину из одной части с какой-то вершиной другой части, то есть не существует ребра, соединяющего две вершины из одной и той же части. Двудольный граф с n вершинами в одной доле и m во второй обозначается Kn,m.



## Теорема Кёнига

Граф G является двудольным тогда и только тогда, когда все [циклы](http://neerc.ifmo.ru/wiki/index.php?title=%D0%9E%D1%81%D0%BD%D0%BE%D0%B2%D0%BD%D1%8B%D0%B5_%D0%BE%D0%BF%D1%80%D0%B5%D0%B4%D0%B5%D0%BB%D0%B5%D0%BD%D0%B8%D1%8F_%D1%82%D0%B5%D0%BE%D1%80%D0%B8%D0%B8_%D0%B3%D1%80%D0%B0%D1%84%D0%BE%D0%B2#def_graph_cycle_1) в графе G имеют чётную длину.

## Доказательство

Рассмотрим двудольный граф. Начнем цикл в доле U. Нужно пройти по четному числу ребер, чтобы вернуться в U снова. Следовательно, при замыкании цикла число ребер будет четным.

Пусть ненулевой граф G [связен](http://neerc.ifmo.ru/wiki/index.php?title=K-%D1%81%D0%B2%D1%8F%D0%B7%D0%BD%D0%BE%D1%81%D1%82%D1%8C) и не имеет циклов нечетной длины. Выберем произвольно вершину u и разобьем множество всех вершин на два непересекающихся множества U и V так, чтобы в U лежали вершины v0, такие что [кратчайшая цепь](http://neerc.ifmo.ru/wiki/index.php?title=%D0%9A%D1%80%D0%B0%D1%82%D1%87%D0%B0%D0%B9%D1%88%D0%B8%D0%B9_%D0%BF%D1%83%D1%82%D1%8C_%D0%B2_%D0%B0%D1%86%D0%B8%D0%BA%D0%BB%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%BE%D0%BC_%D0%B3%D1%80%D0%B0%D1%84%D0%B5) (u,v0) была чётной длины, а в V соответственно вершины v1, для которых длина цепи (u,v1) — нечётная. При этом u ∈ U. В графе G нет ребер ab, таких что a, b лежат одновременно в U и V. Докажем это от противного. Пусть a, b ∈ U. Зададим P0 — кратчайшая (u, a) цепь, а P1 — кратчайшая (u, b) цепь. Обе цепи четной длины. Пусть v0 — последняя вершина цепи P0, принадлежащая P1. Тогда подцепи от u до v0 в P0 и P1 имеют одинаковую длину (иначе бы, пройдя по более короткой подцепи от u до v0 мы смогли бы найти более короткую цепь от u до a или от u до b, чем цепь P0 или P1). Так как подцепи от v0 до a и от v0 до b в цепях P0 и P1 имеют одинаковую четность, а значит в сумме с ребром ab они образуют цикл нечётной длины, что невозможно.

## Следствия

### **Алгоритм проверки графа на двудольность, используя обход в глубину**

Так как граф является двудольным тогда и только тогда, когда все циклы четны, определить двудольность можно за один [**проход в глубину**](http://neerc.ifmo.ru/wiki/index.php?title=%D0%9E%D0%B1%D1%85%D0%BE%D0%B4_%D0%B2_%D0%B3%D0%BB%D1%83%D0%B1%D0%B8%D0%BD%D1%83,_%D1%86%D0%B2%D0%B5%D1%82%D0%B0_%D0%B2%D0%B5%D1%80%D1%88%D0%B8%D0%BD). На каждом шаге обхода в глубину помечаем вершину. Допустим, мы пошли в первую вершину — помечаем её как **1**. Затем просматриваем все смежные вершины, и если не помечена вершина, то на ней ставим пометку **2** и рекурсивно переходим в нее. Если же она помечена и на ней стоит та же пометка, что и у той, из которой шли (в нашем случае **1**), значит граф не двудольный.

### **Алгоритм проверки графа на двудольность, используя обход в ширину**

Произведём серию поисков в [ширину](http://neerc.ifmo.ru/wiki/index.php?title=%D0%9E%D0%B1%D1%85%D0%BE%D0%B4_%D0%B2_%D1%88%D0%B8%D1%80%D0%B8%D0%BD%D1%83). Т.е. будем запускать поиск в ширину из каждой непосещённой вершины. Ту вершину, из которой мы начинаем идти, мы помещаем в первую долю. В процессе поиска в ширину, если мы идём в какую-то новую вершину, то мы помещаем её в долю, отличную от доли текущей вершину. Если же мы пытаемся пройти по ребру в вершину, которая уже посещена, то мы проверяем, чтобы эта вершина и текущая вершина находились в разных долях. В противном случае граф двудольным не является. По окончании работы алгоритма мы либо обнаружим, что граф не двудолен, либо найдём разбиение вершин графа на две доли.

# 47.Числа реберной и вершинной связности графа

Вершинной связностью κ графа G называется наименьшее число вершин, удаление которых приводит к несвязному или тривиальному графу.

Реберной связностью λ графа G называется наименьшее количество ребер, удаление которых приводит к несвязному или тривиальному графу.

Пускай минимальная степень вершины графа G обозначается буквой δ

## Теорема 1

Для любого графа G справедливо следующее неравенство: κ ≤ λ ≤ δ

## Доказательство

Проверим второе неравенство. Если в графе нет ребер, то λ=0. Если ребра есть, то несвязный граф получаем из данного, удаляя все ребра, инцидентные вершине с наименьшей степенью. В любом случае λ ≤ δ

Чтобы проверить первое неравенство нужно рассмотреть несколько случаев.

Если G - несвязный или тривиальный граф, то κ=λ=0

Если G связен и имеет мост x, то λ=1. В последнем случае κ=1, поскольку или граф G имеет точку сочленения, инцидентную ребру x, или же G=K2

Наконец, предположим, что граф G содержит множество из λ≥2 ребер, удаление которых делает его несвязным. Ясно, что удаляя λ−1 ребер из этого множества получаем граф, имеющий мост x = uv. Для каждого из этих λ−1 ребер выберем какую-либо инцидентную с ним вершину отличную от u и v. Удаление выбранных вершин приводит к удалению λ−1 (а возможно, и большего числа) ребер. Если получаемый после такого удаления граф не связен, то κ < λ; если же он связен, то в нем есть мост x, и поэтому удаление вершины u или v приводит либо к несвязному, либо к тривиальному графу. В любом случае κ ≤ λ

## Теорема 2

Для любых натуральных чисел a, b, c, таких что a ≤ b ≤ c, существует граф G, у которого κ=a, λ=b и δ=c

## Доказательство

Рассмотрим граф G, являющийся объединением двух полных графов G1 и G2, содержащих c+1 вершину. Отметим b вершин, принадлежащих подграфу G1 и a вершин, принадлежащих подграфу G2. Добавим в граф G b ребер так, чтобы каждое ребро было инцидентно помеченной вершине, лежащей в подграфе G1 и помеченной вершине, лежащей в подграфе G2, причем не осталось ни одной помеченной вершины, у которой не появилось хотя бы одно новое ребро, инцидентное ей.

Тогда:

Поскольку b ≤ c, то было как минимум две непомеченные вершины, поэтому δ=c, так как минимальные степени вершин графов G1 и G2 были равны c, а степени их вершин не уменьшались.

Заметим, что между двумя вершинами графа G существует не меньше a вершинно-непересекающихся простых цепей, следовательно по [теореме Менгера](https://neerc.ifmo.ru/wiki/index.php?title=%D0%A2%D0%B5%D0%BE%D1%80%D0%B5%D0%BC%D0%B0_%D0%9C%D0%B5%D0%BD%D0%B3%D0%B5%D1%80%D0%B0) κ ≥ a. Однако если удалить из графа G помеченные вершины его подграфа G2, то граф G потеряет связность. Значит, κ = a.

Аналогично рассуждению пункта 2, легко убедится, что λ = b.

# 48. Дерево — связный граф с нулевым цикломатическим числом.

**Цикломатическое число.***Цикломатическое число* графа указывает то число ребер, которое нужно удалить из данного графа, чтобы получить дерево (для связного графа) или лес (для несвязного графа), т.е. добиться отсутствия у графа циклов.

Пусть связный граф *G*(*X, U*) имеет *n=*|*X*| вершин и *r=*|*U*|ребер. Тогда, дерево, построенное на этом графе, имеет |*W*| =*n-*1 ребро. По определению цикломатическое число

***u*(*G*) = *r -*(*n*-1) =*r – n*+1.**

Для несвязного графа с *p*компонентами связности, цикломациктическое число

***u*(*G*) = *r – n*+ *p.***

Цикломатическое число всегда неотрицательно.

Граф G = (V, E) **называется связным**, если для каждой пары вершин v,w ∈ V в графе G есть путь из вершины v в вершину w. Если граф G не является связным, каждый максимальный по включению его связный подграф называется компонентой связности.

**Цикл** – это замкнутый маршрут без повторений ребер.

**Дерево** – это связный граф без циклов.

Пусть Γ – связный граф; выделим в нем произвольный цикл и удалим из этого цикла произвольное ребро. Получим снова связный граф; повторяя эту процедуру, придем в конце концов к связному графу без циклов (дереву). **Определение.** Полученное дерево называется максимальным деревом исходного графа; удаленные ребра называются перемычками, а их число – цикломатическим числом графа или первым числом Бетти.

**Теорема 1.6.** Число вершин **v**, число ребер **e** и первое число Бетти **b** связного графа удовлетворяют соотношению **v − e = 1 − b.**

# 49. Свойства деревьев

**Дерево** - это связный граф без циклов. Граф без циклов называется лесом. Каждая компонента связности леса является деревом.

**Дерево** - это связный граф, в котором при N вершинах всегда ровно N-1 ребро.

**Дерево** - это граф, между любыми двумя вершинами которого существует ровно один путь.

Граф G(V,X) является деревом тогда и только тогда, когда выполняется хотя бы одно из условий:

1. Чтобы простой связный граф был деревом, необходимо и достаточно, чтобы число вершин было больше числа ребер на один.

2. Чтобы граф был деревом, необходимо и достаточно, чтобы любые две вершины его соединялись единственным маршрутом.

3. Граф будет деревом тогда и только тогда, когда добавление любого нового ребра приводит к появлению ровно одного цикла.

4. Граф G(V,X) связен и не содержит циклов.

5. Граф G(V,X) связный , но утрачивает это свойство после удаления любого ребра.

**Свойства:**

1. Дерево не имеет кратных рёбер и петель.
2. Любое дерево с {\displaystyle n}n вершинами содержит {\displaystyle n-1}n-1 ребро. Более того, конечный связный граф является деревом тогда и только тогда, когда {\displaystyle B-P=1}**B – P = 1**, где {\displaystyle B}B — число вершин, {\displaystyle P}P — число рёбер графа.
3. Граф является деревом тогда и только тогда, когда любые две различные его вершины можно соединить единственной простой цепью.
4. Любое дерево однозначно определяется расстояниями (длиной наименьшей цепи) между его концевыми (степени 1) вершинами.
5. Любое дерево является двудольным графом.
6. Любое дерево, множество вершин которого не более чем счётное, является планарным графом.
7. Для любых трёх вершин дерева, пути между парами этих вершин имеют ровно одну общую вершину.

# 50. Валентность вершины графа. Паспорт графа. Теорема о рукопожатиях. Каким может быть паспорт связного графа?

**Степенью вершины** (валентностью) называется число ребер графа, которым принадлежит эта вершина.

**Паспорт графа** – n = (N1, …, Nm), где Ni – валентность вершины.

**Лемма о рукопожатиях**: Сумма степеней вершин равна удвоенному числу ребер.

Следствие 1: Число вершин нечетной степени четно.

Следствие 2: В полном графе m=n\*n/2-n/2. 4. Теорема. Для обыкновенного графа число ребер ограничено.

# 51. Остовное дерево связного графа. Жадный алгоритм.

**Остовным деревом** связного графа G называется любой его подграф D, содержащий все вершины графа G и являющийся деревом.

Минимум остовных деревьев графа**Г** можно найти, применяя процедуру исследования ребер в порядке возрастания их весов. Другими словами, на каждом шаге выбирается новое ребро с наименьшим весом, не образующее циклов с уже выбранными ребрами. Процесс продолжается до тех пор, пока не будет выбрано **|X| - 1** ребро. Рассмотренная процедура называется **жадным алгоритмом.**

# 52. Определение потока на графе. Условия Кирхгофа — линейные уравнения.

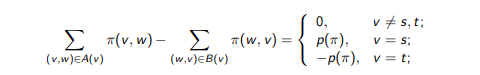
**Сетью** называется ориентированный граф S = (V, E) с одним источником s ∈ V и одним стоком t ∈ V, s 6= t.

**Транспортной сетью** называется связная сеть S = (V, E) с функцией τ : E → R+, сопоставляющей каждой ее дуге e ∈ E неотрицательное действительное число τ (e) ≥ 0, называющееся его пропускной способностью.

**Потоком в транспортной сети** T = (S, τ ) называется такая функция π : E → R+, приписывающая каждой дуге сети e ∈ E неотрицательное число π(e) ≥ 0, называемое потоком по этой дуге, для которой верны свойства:

1) для каждой дуги сети e ∈ E верно π(e) ≤ τ (e), то есть поток по дуге не превышает пропускную способность этой дуги; ццц

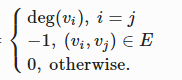
2) для каждой вершины сети v ∈ V верно

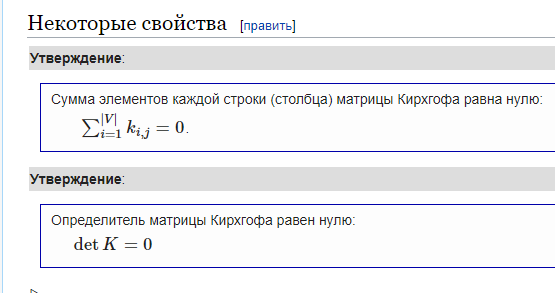


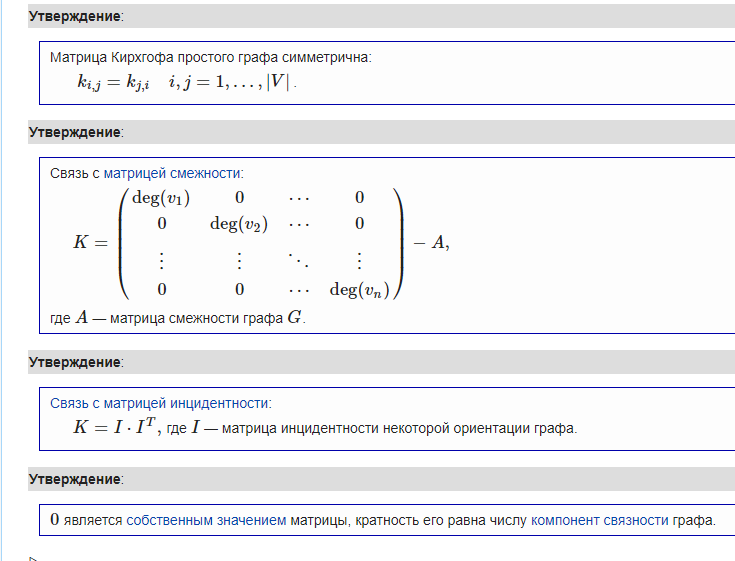
**Закон Кирхгофа** - Сумма всех токов, втекающих в узел, равна сумме всех токов, вытекающих из узла.

# 53.Ранг матрицы уравнений Кирхгофа. Размерность пространства потоков.

**Матрицей Кирхгофа** [простого графа](https://neerc.ifmo.ru/wiki/index.php?title=%D0%9E%D1%81%D0%BD%D0%BE%D0%B2%D0%BD%D1%8B%D0%B5_%D0%BE%D0%BF%D1%80%D0%B5%D0%B4%D0%B5%D0%BB%D0%B5%D0%BD%D0%B8%D1%8F_%D1%82%D0%B5%D0%BE%D1%80%D0%B8%D0%B8_%D0%B3%D1%80%D0%B0%D1%84%D0%BE%D0%B2#def_simple_graph) G=(V,E) называется матрица   элементы которой определяются равенством:

ki,j = 





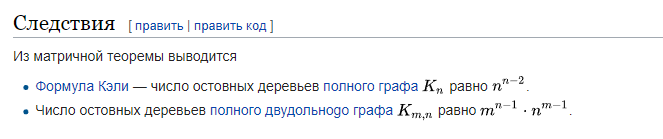
Пусть KK — матрица Кирхгофа графа GG, I — матрица инцидентности GG с некоторой ориентацией. Тогда K=I⋅I(транспонированная).

# 54. Базис пространства потоков, связанный с остовным деревом.

Пусть {\displaystyle G}G — связный помеченный граф с матрицей Кирхгофа {\displaystyle M}M. Все алгебраические дополнения матрицы Кирхгофа {\displaystyle M}M равны между собой и их общее значение равно количеству остовных деревьев графа {\displaystyle G}G.

Пусть HH — обыкновенный (n,n−1)(n,n−1) — граф, n⩾2n⩾2, II — матрица инцидентности некоторой его ориентации, MM — произвольный минор порядка n−1n−1 матрицы II. Тогда

1. если HH не является деревом, то M=0M=0;
2. если HH — дерево, то M=±1M=±1.



### Следствие

Число остовов в полном графе Kn равно nn-2.

# 55. Теорема Кирхгофа о том, на каком наборе ребер можно задавать значения потока для его однозначного определения.

# 56. Градиент функции, заданной на вершинах графа. Размерность пространства градиентов. Разрез графа.

# 57. Ортогональность пространства потоков и пространства градиентов.