1. Boolean function

Булева фукнция – это фукнция из n-ой степени булева множества в булево множество, где n – это арность функции, булево множество {0,1} или {F,T} – домен функции

2. Boolean formula

Булева формула – представление булевой функции в виде формулы, которая может содержать логические переменные и операции между ними (конъюнция, дизъюнция, импликация, отрицание и т.д.)

3. Boolean circuit

Булева схема – это модель, состоящая из логических элементов, которые принимают на вход переменные, а выход – это значение булевой функции, которой соотвествует этот элемент. Логические элементы: and, or, nand, nor

4. Number of n-ary Boolean functions

2^(2^n), где n – это арность функции. Т.к. каждая переменная в векторе может принимать два значения – 0 или 1, а в векторе их n, значит общее кол-во векторов 2^n, а каждый может принимать еще 2 значения

5. Minterm and maxterm

Минтерм – это функция, принимающее значение 1 только на единственном наборе. Представляет собой конъюнкцию литералов, где каждая переменная встречается 1 раз

Макстерм – это функция, принимающая значение 0 только на единственном наборе, дизъюнкция литералов, где каждая переменная встречается 1 раз

6. DNF and СNF

Дизъюнктивная нормальная форма – это формула, которая представляет собой дизъюнкцию кубов

CNF – это формула, которая представляет собой конъюнкцию клоз

7. CDNF and CCNF

Канонические формы – дизъюнкция минтермов или конъюнкция макстермов соответственно

8. Negation normal form

Формула, состоящая из трех операций: отрицание (применимо только к переменным), конъюнкция и дизъюнкция

9. Blake canonical form

Формула – дизъюнкция всех простых импликант

10. Algebraic normal form

Полином Жегалкина – формула, состоящая из кубов, связанных операцией xor

11. Binary decision diagram

Бинарная диаграмма решений – форма представления булевой функции в виде ориентированного ациклического графа, который состоит из узлов решений, каждый из которых имеет по два потомка – переход в левого означает, что переменная, из которой выходит, принимает значение 0, а в правого – 1 – и конечных узлов, принимающих значение 0 или 1 – результат функции.

12. Shannon expansion

Разложение Шеннона по переменной x – форма представления булевой функции от n переменных в виде дизъюнкции конъюнкций этой переменной и функции от n-1 переменных, где вместо переменной x ставится 1, и отрицания этой переменной и функции от n-1 переменных, где вместо переменной x ставится 0

13. Implicant

Импликанта булевой функции f – это булева функция g, которая удовлетворяет свойству: если на каком-то наборе g принимает значение 1, то f тоже принимает значение 1

14. Prime implicant

Простая импликанта – это импликанта k, для которой не существует другой импликанты, которая поглощает k (наименьшая по включению литералов, наибольшая по включению минтермов)

15. Essential prime implicant

Это простая импликанта, которая включает в себя такой минтерм, который не покрывается ни одной другой импликантой. Такие простые импликанты всегда входят в построение минимальной ДНФ

16. Composition(superposition) of Boolean functions

Суперпозиция булевых функций – это функция, полученная из некоторого множества функций путем подстановки одной функции в другую или отождествления переменных

17. Closed set

Множество является замкнутым относительно какой-то операции, если результат этой операции, которая применяется к любым элементам множества, лежит в этом множестве.

18. Closure

Замыканием множества относительно операции называется минимальное множество, которое является замкнутым относительно этой операции

19. Functional completeness

Функциональная полнота – это свойство набора функций, когда с помощью операции суперпозиции из этого набора можно выразить любую булеву функцию

20. Post’s criterion

Критерий Поста – с помощью него можно определить, является ли набор функционально полным. Набор является функционально полным, если в наборе существует хотя бы одна функция для каждого свойства, которая ему удовлетворяет:

Не сохраняет единицу, не сохраняет 0, не является линейной, монотонной и самодвойственной

21. Post’s classes

1) Сохраняет единицу: на единичном наборе принимает значение 1

2) Сохраняет ноль: на нулевом наборе принимает значение 0

3) Является самодвойственной: функция от противоположного набора противоположна функции от набора

4) Является линейной: полином Жегалкина линеен

5) Является монотонной: для любого возрастающего набора функция не убывает

22. Post’s lattice

23. Number of n-ary truth- and falsity- preserving functions

2^(2^n – 1)

24. Number of n-ary self-dual functions

2^(2^(n-1))

25. Number of n-ary monotonic functions

Число дедекинда

26. Number of n-ary linear functions

2^(n+1)

27. Construction of Zhegalkin polynomial

Метод треугольника, метод Паскаля

28. Gray Code

29. Boolean satisfiability problem

Задача выполнимости булевых формул – это задача, которая заключается в следующем: есть какая-то функция f от какого-то количества аргументов. Нужно найти подстановку значений в эту функцию, чтобы она стала истинной. Не всегда для функции есть таблица истинности, она может быть задана формулой.

Просто SAT – ответить на вопрос (decision problem), случай называется SAT или UNSAT

Functional SAT – найти подстановку (model), или доказать, что ее нет (proof, сертификат не выполнимости), или количество подстановок (sharp-SAT)

Обычно имеется в виду CNF-SAT, то есть функция задана в виде КНФ формулы

30. Polynomial algorithm for 2-SAT

2-SAT – частный случай SAT, когда функция задана в виде CNF, в которой все клозы состоят из двух литералов (формулы Крома). Ключевым понятием для алгоритма является граф импликации. Если рассмотреть каждую 2-клозу, то она будет эквивалентна двум импликациям, каждые из которых задают ребро в графе. Для каждой клозы составляем две импликации. Каждая импликация рисуется направленным ребром. В графе рассматриваем все компоненты сильной связности (подграф, в котором из каждой вершины можно дойти до любой другой, классы эквивалентности). Анализируем все компоненты и если в какой-то компоненте встретились переменная с ее отрицанием, то решения нет. Если такого случая нет, то задача выполнима

31. DPLL

Алгоритм для решения CNF-SAT, заключается в следующем: делается выбор переменной, подстановка значений и упрощение (unit propagation). Дальше, если нашлось противоречие, делается backtrack(переход на предыдущую стадию, выбор другой переменной).

32. CDCL

33. Complexity classes P and NP

Пример: 3-SAT. P – класс полиномиально разрешимых задач (задач, у которых есть какой-то алгоритм, который решает ее за полином). NP – не детерминированно полиномиальные задачи (задачи, решаемые за полиномиальное время на мистической недетерминированной машине Тьюринга). Машина Тьюринга – это универсальный вычислитель, модель вычислений, эквивалентная тому, что есть у человечества с точки зрения вычислений. Детерминированность означает, что мы можем точно определить, как будет вести себя вычислитель при определенных входных данных.

34. NP-hardness

NP-hard задачи – это задачи, к которым можно свести все NP-задачи за полиномиальное время. Чтобы доказать, что задача NP-hard, можно свести SAT к ней или любую NP-complete задачу. Сведение по Карпу

35. NP-complete problems

NP-complete задачи – это NP-hard, которые сами являются NP. Пример: SAT, задача о раскраске графа, задача о рюкзаке

36. Proposition

Высказывание – это утверждение, которое может быть правдивым либо ложным

37. Language of propositional logic

Алфавит состоит из атомов (несократимых формул без логических связок: переменные – заглавные латинские буквы, могут быть с индексами; константы – true и false) и операторов (отрицание, конъюнкция, дизъюнкция, xor, импликация, эквивалентность). Предложения – 1) любой атом 2) отрицание переменной 3) если A и B предложения, то все бинарные операции тоже предложения 4) больше ничего не является предложением

38. Logical argument

Логический аргумент – это набор предположений (премисов), из которых следует утверждение (вывод)

39. Interpretation

Интерпретация – присвоение переменным значения true или false

40. Tautology and contradiction

A – тавтология, если оно верно для любой подстановки значений. А – противоречие, если ложно для всех подстановок значений

41. Satisfiable and falsifiable statements

A satisfiable – верно для какой-то подстановки значений. A falsifiable – ложно для какой-то подстановки значений.

42. Semantic entailment

Логическое следствие – из набора утверждений А1…Аn следует С, если нет такой подстановки значений, для которой все утверждения А1..Аn верны, а С ложно

43. Soundness

Каждое доказуемое утверждение в действительности правда

44. Completeness

Каждое правдивое утверждение имеет доказательство

45. Natural deduction

Система доказательств, в которой логические рассуждения выражаются с помощью правил вывода

46. Fitch notation

Система обозначений для формальных доказательств

47. Modus ponens

Правило вывода: если A и A->B выводимые формулы, то B также выводима

48. Prefix code

Ни одно кодовое слово не является началом другого

49. Shannon-Fano coding

50. Huffman coding

51. Arithmetic coding

52. Hamming code