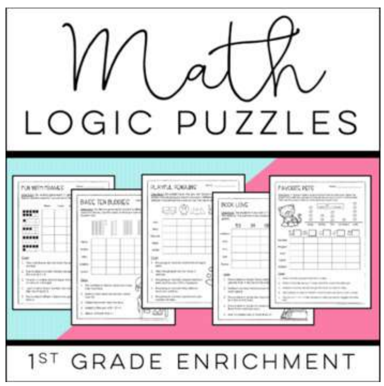
Подготовка к экзамену по дисциплине «Математическая логика и теория алгоритмов»

Преподаватель: Моргунов В. М
Студент: Дьяченко С. С

22 января 2024 г.



1 Понятие высказывания

Определение: Высказывание - это простое утдвердительное предложение (содержит одну мысль), о котором можно сказать истинно они или ложно.
Пример: Люди сметрны!

2 Логическое значение высказывания

Факт: Существуют два значения для высказываний — «истина» (1) и «ложь» (0).

$$\lambda(p) = \begin{cases} 0, \text{ если } p- \text{ ложно} \\ 1, \text{ если } p- \text{ истинно} \end{cases}$$

3 Классификация высказываний

Определение: Тавтология - высказывание принимающее логическое значение 1 при любом наборе значений пропозициональных переменных.
Определение: Тождественная ложь - высказывание принимающее логическое значение 0 при любом наборе значений пропозициональных переменных.
Определение: Выполнимое - высказывание принимающее логическое значение 1 хотя бы на одном наборе значений пропозициональных переменных.
Определение: Опровержимое - высказывание принимающее логическое значение 0 хотя бы на одном наборе значений пропозициональных переменных.

4 Понятие логической функции

Определение: Логическая функция — это отображение множества высказываний в множество логических значений.
Логические операции позволяют построить сложные высказывания из данных высказываний, при котором истинностное значение сложного высказывания полностью определяется значениями исходных высказываний.
Частный случай: Логические связки - булевые функции.

5 Таблицы истинности логические функции

Определение: Таблица истинности — это таблица, которая показывает значения логической функции в зависимости от значений ее аргументов (какое булево значение будет возвращено функцией при различных комбинациях значений ее аргументов).
Пример:

A	B	A $\circ$ B
0	0	1
0	1	1
1	0	0
1	1	0

6 Основные логические функции

A B	Функция	Пример	Название	0 0	0 1	1 0	1 1
1	$\neg$	$\neg A$	Инверсия	1		0	
2	$\vee$	$A \vee B$	Конъюнкция	0	1	1	1
3	$\wedge$	$A \wedge B$	Дизъюнкция	0	0	0	1
4	$\rightarrow$	$A \rightarrow B$	Импликация	1	1	0	1
5	$\leftrightarrow$	$A \leftrightarrow B$	Эквиваленция	1	0	0	1
6	$\oplus$	$A \oplus B$	по модулю 2	0	1	1	0
7	/	A/B	Шеффера штрих	1	1	1	0
8	$\downarrow$	$A \downarrow B$	Стрелка Пирса	1	0	0	0

7 Законы алгебры высказываний

Определение: Алгебра высказываний: $A = \{P, \Phi\}$, где $P = \{\ \neg, \vee, \wedge, \rightarrow, \leftrightarrow\ \}$ - булевые связки, а $\Phi-$ алфавит высказываний (пропозициональных переменных.
Законы:

Идемпотентность	A ∨ A = A	A ∧ A = A
Коммутативность	A ∨ B = B ∨ A	A ∧ B = B ∧ A
Ассоциативность	(A ∨ B) ∨ C = A ∨ (B ∨ C)	(A ∧ B) ∧ C = A ∧ (B ∧ C)
Дистрибутивность	A ∧ (B ∨ C) = (A ∧ B) ∨ (A ∧ C)	A ∨ (B ∧ C) = (A ∨ B) ∧ (A ∨ C)
Действия с константами	A ∨ 1 = 1 A ∨ 0 = A	A ∧ 1 = A A ∨ 0 = 0
Исключенного третьего	¬ A ∨ A = 1	
Двойного отрицания	¬(¬A) = A	
Противоречия		¬A ∧ A = 0
Де Моргана	¬(A ∨ B) = ¬A ∨ ¬ B	¬ (A ∧ B) = ¬A ∨ ¬B

8 Понятие силлогизма

Определение: Тавтологии представляют собой схемы построения истинных высказываний, независимо от содержания и истинности составляющих высказываний. Основное значение тавтологий состоит в том, что некоторые из них предоставляют правильные способы построения умозаключений, т.е. такие способы, которые от истинных посылок всегда приводят к истинным выводам.
Определение: Силлогизм — это цепочка (форма логического вывода), основанная на трех законах: законе тождества, законе противоречия и законе

исключенного третьего.

$$(A \to B) \land (B \to C) \to (A \to C)$$

Задействует логическое следование: $\overbrace{(A \to B), (B \to C)}^{assumptions} \vdash \overbrace{(A \to C)}^{consequence}$, *assumptions* - посылки, *consequence* - следствие.
Пример: "Все люди смертны, следовательно, Сократ смертен."
Здесь "Все люди смертны" - посылка и "Сократ - человек" - посылка, а "Сократ смертен" — следствие.
Определение: Энтимема - силлогизм с пропущенной посылкой или заключением.
Определение: Эпихейрема – сложносокращенный силлогизм обе посылки которого – сокращенные силлогизмы (энтимемы).
соединение сокращённых силлогизмов, в которых опущена или большая, или меньшая посылка.
Определение: Полисиллогизм - два или более простых силлогизма, которые связаны между собой так, что вывод одного из них служит посылкой другого.
Пример: (Все продукты, содержащие витамины, полезны. Все фрукты содержат витамины.)
(Все фрукты полезны. Яблоко - это фрукт.) Яблоки полезны.
Определение: Сорит - это сложносокращенный силлогизм; полисиллогизм с пропущенной посылкой последующего силлогизма, являющейся выводом предыдущего силлогизма.
Пример: (Все продукты, содержащие витамины, полезны. Все фрукты содержат витамины.)
(*(пропущен вывод, что все фрукты полезны)* Яблоко - это фрукт.) Яблоки полезны.

9 Понятие контрапозиции

Определение: Тавтологии представляют собой схемы построения истинных высказываний, независимо от содержания и истинности составляющих высказываний. Основное значение тавтологий состоит в том, что некоторые из них предоставляют правильные способы построения умозаключений, т.е. такие способы, которые от истинных посылок всегда приводят к истинным выводам.
Определение: Контрапозиция — это закон, логический инструмент, который заключается в отрицании какого-либо утверждения или явления. Позволяющих с помощью отрицания менять местами посылку и следствие высказывания.

$$A \to B \vdash \neg B \to \neg A$$

Пример: "Без огня нет дыма", значит "Если есть дым - есть огонь"

10 Понятие формулы логики высказываний ЛВ

Определение: Алгебра высказываний: $A = \{P, \varPhi\}$, где $P = \{\ \neg,\ \vee,\ \wedge,\ \rightarrow,\ \leftrightarrow\ \}$ - булевые связки, а $\varPhi-$ алфавит высказываний (пропозициональных переменных.

Определение: Формула такой логики(алгебры) - высказывание, составленное из элементов алфавита: пропозициональных переменных, логических связок и скобок.

Справедливо:

- Всякая пропозициональная переменная есть формула.
- Если p и q - формулы, то выражения $(\neg p), (p \vee q), (p \wedge q), (p \rightarrow q), (p \leftrightarrow q)$ - тоже формулы
- Других формул не существует

Пример: $(A \rightarrow B) \rightarrow (B \rightarrow C)$

11 Существенные и фиктивные переменные формулы ЛВ

Определение: Функция $F(X_1, X_2 \ldots X_{i-1}, X_i, X_{i+1} \ldots X_n)$ существенно зависит от переменной X_i, если у переменных $X_1 \ldots, X_{i-1}, X_{i+1} \ldots X_n$ существует такой набор значений $\alpha_1, \alpha_2 \ldots \alpha_{i-1}, \alpha_{i+1} \ldots \alpha_n$, что выполняется неравенство:

$$F(\alpha_1, \alpha_2 \ldots \alpha_{i-1}, 0, \alpha_{i+1} \ldots \alpha_n) \neq F(\alpha_1, \alpha_2, \ldots \alpha_{i-1}, 1, \alpha_{i+1} \ldots \alpha_n)$$

Определение: Переменную X_i называют фиктивной, если для любого набора значений $\alpha_1, \alpha_2 \ldots \alpha_{i-1}, \alpha_{i+1} \ldots \alpha_n$ переменных $X_1, X_2 \ldots X_{i-1}, X_{i+1} \ldots X_n$ выполняется равенство:

$$f(\alpha_1, \alpha_2 \ldots \alpha_{i-1}, 0, \alpha_{i+1} \ldots \alpha_n) = f(\alpha_1, \alpha_2, \ldots \alpha_{i-1}, 1, \alpha_{i+1} \ldots \alpha_n)$$

Существенные переменные влияет на значение формулы ЛВ, фиктивная - нет.

12 Теорема о признаке равносильности формул ЛВ

Определение: Тавтологии представляют собой схемы построения истинных высказываний, независимо от содержания и истинности составляющих высказываний.

Определение: Формулы $F(X_1, X_2, \ldots, X_n)$ и $H(X_1, X_2, \ldots, X_n)$ алгебры высказываний равносильны(эквивалентны), если при любых значениях пропозициональных переменных логические значения формул F и H совпадают. Для указания равносильности формул используют $F \cong H$. Равносильность формул для любых конкретных высказываний $A_1, A_2, \ldots, A_n$:

$$F \cong H \quad \leftrightarrow \quad \lambda\big(F(A_1, A_2, \ldots, A_n)\big) = \lambda\big(H(A_1, A_2, \ldots, A_n)\big)$$

Теорема: О признаке равносильности формул: Две формулы F и H алгебры высказываний равносильны тогда и только тогда, когда формула $F \leftrightarrow H$ является тавтологией:

$$F \cong H \quad \Leftrightarrow \quad \vDash F \leftrightarrow H$$

Доказательство: Если $F \cong H$, то по определению $\lambda\big(F(A_1, \ldots, A_n)\big) = \lambda\big(H(A_1, \ldots, A_n)\big)$ для любых высказываний $A_1, \ldots, A_n$. Тогда $\lambda\big(F(A_1, \ldots, A_n)\big) \leftrightarrow \lambda\big(H(A_1, \ldots, A_n)\big) = 1$, откуда на основании $\lambda(P \leftrightarrow Q) = \lambda(P) \leftrightarrow \lambda(Q)$, $\lambda\big(F(A_1, \ldots, A_n)\big) \leftrightarrow \lambda\big(H(A_1, \ldots, A_n)\big) = 1$ для любых $A_1, \ldots, A_n$. Последнее означает по определению тавтологии, что $\vDash F \leftrightarrow H$. Обратными рассуждениями доказывается утверждение: если $\vDash F \leftrightarrow H$, то $F \cong H$.

13 Понятие логического следования

Определение: Формула G - логическое следствие формул $F_1, F_2 \ldots F_m$, если она превращается в истинное высказывание при всякой такой подстановке вместо всех ее пропозициональных переменных $X_1 \ldots X_n$ конкретных высказываний, при которой в истинное высказывание превращаются все формулы $F_1(X_1 \ldots X_n), \ldots, F_m(X_1 \ldots X_n)$. Логическое следствие обозначается так: $F_1, \ldots, F_m \vDash G$. Формулы $F_1 \ldots F_m$ называются посылками для логического следствия G.

$$F_1, F_2, F_3 \cdots F_n \models G, \text{только если } \Big(\lambda(F_1) = \lambda(F_2) = \cdots = 1 \rightarrow \lambda(G) = 1\Big)$$

Пример: Если я $\overset{F_1}{\text{предварительно получу 4}}$ и $\overset{F_2}{\text{выучу билеты}}$,
то $\overset{G}{\text{буду сдавать экзамен по МЛТА}}$

14 Теорема о признаке логического следования

Теорема: о признаке логического следствия: Формула G будет логическим следствием формулы F тогда и только тогда, когда формула $F \rightarrow H$ является тавтологией:

$$F \vDash H \Leftrightarrow \quad \vDash F \rightarrow H$$

Доказательство: Необходимость. Дано: $F(X_1,\dots,X_n) \vDash H(X_1,\dots,X_n)$, т.е. если для набора высказываний $A_1,\dots,A_n$ имеет место $\lambda\big(F(A_1,\dots,A_n)\big) = 1$, то $\lambda\big(H(A_1,\dots,A_n)\big) = 1$. Тогда для любого набора высказываний $A_1,\dots,A_n$ имеет место равенство:

$$\lambda\big(F(A_1,\dots,A_n)\big) \to \lambda\big(H(A_1,\dots,A_n)\big)$$

Так как равенство не может быть равно нулю и $\lambda\big(F(A_1,\dots,A_n)\big) \to \lambda\big(H(A_1,\dots,A_n)\big) = \lambda\big(F(A_1,\dots,A_n) \to H(A_1,\dots,A_n)\big) = 1$, то $\lambda\big(F(A_1,\dots,A_n) \to H(A_1,\dots,A_n)\big) = 1$ для любых высказываний $A_1,\dots,A_n$. Это означает, что формула $F(X_1,\dots,X_n) \to H(X_1,\dots,X_n)$ — тавтология, т.е. $\vDash F \to H$.
Доказательство: Достаточность. Из необходимости:

$$\lambda\big(F(A_1,\dots,A_n)\big) \to \lambda\big(H(A_1,\dots,A_n)\big) = 1$$

Предположим теперь, что $\lambda\big(F(A_1,\dots,A_n)\big) = 1$. Тогда: $1 \to \lambda\big(H(A_1,\dots,A_n)\big) = 1$, откуда $\lambda\big(H(A_1,\dots,A_n)\big) = 1$, ибо в противном случае $1 \to 0 = 1$ — противоречие. Это значит (по определению логического следствия), что $F \vDash H$.

15 Свойства логического следования

Факт: Свойства логического следования между формулами алгебры высказываний:

- 1) $F_1, F_2, \dots, F_m \vDash F_i$ для $i = 1 \dots m$ - отношение логического следования рефлексивно
- 2) Если $F_1, F_2 \dots F_m \vDash G_i$ и $G_1, G_2 \dots G_p \vDash H$, то $F_1, F_2 \dots F_m \vDash H$

16 Формализованное исчисление высказываний (ФИВ)

Определение: Язык ФИВ: пропозициональные переменные, логические связки $(\neg, \to)$, (,) — технические знаки.
Определение: Формулы ФИВ (как в алгебре высказываний):

- каждая пропозициональная переменная есть формула
- если $F_1 F_2$ - формулы, то выражения $\neg F_1, (F_1 \to F_2)$ тоже формулы
- никаких других формул нет

Определение: Система аксиом ФИВ:

- **(A1)** $F \to (G \to F)$
- **(A2)** $\big(F \to (G \to H)\big) \to \big((F \to G) \to (F \to H)\big)$

- **(A3)** $(\neg G \to \neg F) \to \big((\neg G \to F) \to G\big)$

Определение: Правило вывода - правило отсечения(*modus ponens*):

$$\frac{F, F \to G}{G}$$

$F \wedge G \cong \neg(F \to \neg G)$
$F \vee G \cong \neg F \to G$

17 Свойства ФИВ

Факт: Полнота, непротиворечивость, разрешимость.
Теорема: О полноте: Формула тогда и только тогда доказуема в формализованном исчислении высказываний, когда она является тавтологией алгебры высказываний:

$$\vdash F \Leftrightarrow \vDash F$$

Теорема: О полноте: Всё, что истинно, то выводимо, и обратно:

$$\vDash F \ \Leftrightarrow \vdash F$$

Теорема: О непротиворечивости: F и $\neg F$ - не могут одновременно быть теоремами данной аксиоматическиой теории, $\big(\vdash F \ \ \Leftrightarrow \nvdash (\neg F)\big)$ - F тогда выводима, когд $\neg F$ не выводима.
Теорема: О разрешимости: ФИВ - разрешимая аксиоматическая теория - существует алгоритм, позволяющий для любого утверждения, сформулированного в терминах теории, ответить на вопрос, будет или нет это утверждение теоремой данной теории.

18 Понятие вывода (доказательства) формулы ФИВ

Определение: Вывод (доказательство) **F** из множества формул $\mathbf{\Gamma}$(множества гипотез, посылок вывода) - это такая конечная последовательность $B_1, B_2 \dots B_s$ формул, что каждая её формула - либо аксиома, либо формула из $\mathbf{\Gamma}$, либо получена из двух предыдущих формул этой последовательности по правилу MP, при этом последняя формула $=$ **F**.
Если есть вывод формулы **F** из множества $\mathbf{\Gamma}$,то **F** выводима из $\mathbf{\Gamma}$: $\Gamma \vdash F$.
Если есть вывод формулы **F** из пустого множества гипотез, что **F** выводима из аксиом $\vdash F$ - теорема.

Пример: $\vdash \ F \to F$

$$
\begin{aligned}
&(1): \quad F \to ((F \to F) \to F) && \text{(A1)}\\
&(2): \quad \big(F \to ((F \to F) \to F)\big) \to \big((F \to (F \to F)) \to (F \to F)\big) && \text{(A2)}\\
&(4): \quad \big(F \to (F \to F)\big) \to (F \to F) && \text{(MP)}\\
&(4): \quad F \to (F \to F) && \text{(A1)}\\
&(5): \quad F \to F. && \text{(MP)}
\end{aligned}
$$

19 Теорема о дедукции

Теорема: О дедукции: Если $F_1 \dots F_{m-1}, F_m \vdash G$, то $F_1 \dots F_{m-1} \vdash F_m \to G$. В частности, если $F \vdash G$, $\vdash F \to G$.

20 Понятие предиката

Определение: Определенный на множествах $M_1, M_2 \dots M_n$ n-местным предикат - это предложение, содержащее n переменных $x_1, x_2 \dots x_n$, превращающееся в высказывание при подстановке вместо этих переменных любых конкретных элементов из множеств $M_1, M_2 \dots M_n$ соответственно.
Пример: "Действительное число x кратно 2378"

21 Множество истинности предиката

Определение: Множество истинности предиката P заданного на множествах $M_1, M_2 \dots M_n$ - это совокупность всех значений таких, что данный предикат $P(a_1, a_2, \dots, a_n)$ обращается в истинное высказывание. Это множество M^+:

$$M^+ = \big\{(a_1, a_2, \dots, a_n) \colon \ \lambda\big(P(a_1, a_2, \dots, a_n)\big) = 1\big\}$$

22 Классификация предикатов

Определение: Предикат P, заданный на множествах $M_1, M_2 \dots M_n$:

- Тождественная истинна (ложь), если при любой подстановке вместо переменных $x_1, x_2, \dots, x_n$ любых конкретных предметов $a_1, a_2, \dots, a_n$ из множеств $M_1, M_2, \dots, M_n$ его логическое значение 1 (0)

- Выполнимый (опровержимый), если существует хотя бы один набор конкретных предметов $a_1, a_2 \dots a_n$ из множеств $M_1, M_2 \dots M_n$, при подстановке которых вместо соответствующих предметных переменных предикат P принимает логическое значение 1 (0).

- Общезначимый - тождественная истинна на всякой области, на любой модели.

23 Понятие формулы логики предикатов (ЛП)

Определение: Алфавит формулы: предметные переменные; нульместные предикатные переменные: P, Q, R, P_i, Q_i, R; n-местные ($n \geqslant 1$) предикатные переменные: $R(, \dots,), P_i(, \dots,)$; символы логических операций: $\neg,\ \wedge,\ \vee,\ \to,\ \leftrightarrow$; кванторы: $\forall, \exists$; вспомогательные символы: (,).
Определение: Формула логики предикатов - сложное предложение определённое на множестве, составленное из элементов алфавита:
Справедливо:

- Каждая нульместная предикатная переменная есть формула.
- Если $P(\ , \dots,)$ — n-местная предикатная переменная, то $P(x_1, \dots, x_n)$ есть формула, в которой все предметные переменные $x_1, \dots, x_n$ свободны.
- Если F_1, F_2 — формулы и если предметные переменные, входящие одновременно в обе эти формулы, свободны в каждой из них, то выражения:$\neg F$,$(F_1 \wedge F_2)$, $\quad (F_1 \vee F_2)$, $\quad (F_1 \to F_2)$, $\quad (F_1 \leftrightarrow F_2)$ - тоже формулы.
- Если F — формула и x — предметная переменная, входящая в F свободно, то выражения $(\forall x)(F)(\exists x)(F)$ также являются формулами, в которых переменная x связанная, а все остальные предметные переменные, входящие в формулу F свободно или связанно, остаются и в новых формулах соответственно такими же.
- никаких других формул логики предикатов нет.

Пример:
$$(\forall a \neg P(a) \wedge \exists x(P(x) \wedge Q(x, y)))$$

24 Свободные и связанные переменные формулы ЛП

Определение: Свободные переменные формулы ЛП - не входит в область действия квантора по этой переменной. - не применена кванитификация.
Определение: Связанная переменные формулы ЛП - это переменные, над которыми применена кванитификация.
Пример:
$$\forall x\big[(\exists y)\big(P(x, y)\big) \to Q(x, y, z)\big]$$
x - связанная
y - частично связанная
z - свободная

25 Теорема о приведенной форме для формулы ЛП

Определение: Приведенной формой для формулы логики предикатов называется такая равносильная ей формула, в которой из операций алгебры высказываний имеются только операции $\neg$, $\wedge$, $\vee$, причем знаки отрицания относятся лишь к предикатным переменным и к высказываниям.
Теорема: Для каждой формулы логики предикатов существует приведенная форма.
Доказательство: Проводтся методом математической индукции по числу логических связок в формуле (включая кванторы общности и существования).

26 Теорема о предварённой нормальной форме для формулы ЛП

Определение: Предваренной нормальной формой для формулы логики предикатов называется такая ее приведенная форма, в которой все кванторы стоят в начале, а область действия каждого из них распространяется до конца формулы, т. е. это формула вида $(K_1x_1)\ldots(K_mx_m)\big(F(x_1,\ldots,x_n)\big)$, где K_i есть один из кванторов $\forall$ или $\exists(i=1,\ldots,m)$, $m \leqslant n$, причем формула F не содержит кванторов и является приведенной формулой. (Заметим, что кванторы в формуле могут отсутствовать вовсе.)
Теорема: Для каждой формулы логики предикатов существует предваренная нормальная форма.
Доказательство: Проводтся по индукции, следуя правилу построения формул логики предикатов.

27 Кванторные законы логики предикатов

1. Законы де Моргана для кванторов:
1.1 $\neg(\forall x)(P(x)) \leftrightarrow (\exists x)(\neg P(x))$
1.2 $\neg(\exists x)(P(x)) \leftrightarrow (\forall x)(\neg P(x))$
2. Выражение кванторов одного через другой
2.1 $(\forall x)(P(x)) \leftrightarrow \neg(\exists x)(\neg P(x))$
2.2 $(\exists x)(P(x)) \leftrightarrow \neg(\forall x)(\neg P(x))$
3. Пронесение кванторов через конъюнкцию и дизъюнкцию
3.1 $(\forall x)(P(x) \wedge Q(x)) \leftrightarrow (\forall x)(P(x)) \wedge (\forall x)(Q(x))$
3.2 $(\exists x)(P(x) \vee Q(x)) \leftrightarrow (\exists x)(P(x)) \vee (\exists x)(Q(x))$
3.3 $(\forall x)(P(x) \vee Q) \leftrightarrow (\forall x)(P(x)) \vee Q$
3.4 $(\exists x)(P(x) \wedge Q) \leftrightarrow (\exists x)(P(x)) \wedge Q$
4. Пронесение кванторов через импликацию предикатов
4.1 $(\forall x)(P(x) \to Q) \leftrightarrow ((\exists x)(P(x)) \to Q)$

4.2 $(\exists x)(P(x) \to Q) \leftrightarrow ((\forall x)(P(x)) \to Q)$
4.3 $(\forall x)(Q \to P(x)) \leftrightarrow (Q \to (\forall x)(P(x)))$
4.4 $(\exists x)(Q \to P(x)) \leftrightarrow (Q \to (\exists x)(P(x)))$
5. Законы коммутативности для кванторов
5.1 $(\forall x)(\forall y)(P(x,y)) \leftrightarrow (\forall y)(\forall x)(P(x,y))$
5.2 $(\exists x)(\exists y)(P(x,y)) \leftrightarrow (\exists y)(\exists x)(P(x,y))$
5.3 $(\exists y)(\forall x)(P(x,y)) \to (\forall x)(\exists y)(P(x,y))$
6. Законы универсальной конкретизации и экзистенциального обобщения
6.1 $(\forall x)(F(x)) \vDash F(y)$
6.2 $F(y) \vDash (\exists x)(F(x))$

28 Формализованное исчисление предикатов (ФИП)

Определение: Язык ФИП (алфавит исчисления предикатов): предметных переменных $x_1, x_2 \ldots$, предметных констант (символы выделенных элементов) $c_1, c_2 \ldots$, предикатных букв$P_1', P_2' \ldots P_k' \ldots$, функциональных букв $f_1'', f_2'' \ldots f_\ell'' \ldots$, а также знаков логических связок $\neg\wedge$, кванторов $\forall$, $\exists$ и скобок (,). При этом верхние индексы предикатных и функциональных букв указывают число аргументов соответственно предиката или функции, которые могут быть подставлены вместо этих букв.
Определение: Формулы ФИП (как в логике предикатов): Сначала определяются термы. Ими являются отдельно взятые предметные переменные и константы, а также выражения вида $f^n(t_1, \ldots, t_n)$, где f^n — n-местный функциональный символ; $t_1 \ldots t_n$ — термы
Справедливо:

- если P_n — предикатная буква, $t_1 \ldots t_n$ — термы, то $P_n(t_1 \ldots t_n)$ — формула; при этом все вхождения переменных в эту формулу называются свободными

- если F_1, F_2 — формулы, то формулами являются $\neg F_1$, $(F_1 \to F_2)$; причем все вхождения переменных, свободные в F_1, F_2, являются свободными и в формулах указанных видов; кроме того, можно считать, что в F_1 и F_2 нет предметных переменных, которые связаны в одной формуле и свободны в другой

- если F_x — формула, содержащая свободные вхождения переменной x, то $(\forall x)(F(x))$ и $(\exists x)(F(x))$ — формулы; при этом вхождения переменной x в них называются связанными; вхождения же всех остальных предметных переменных в эти формулы остаются свободными (связанными), если они были свободными (связанными) в формуле $F(x)$ (формула $F(x)$ называется областью действия квантора)

- никаких других формул нет

Определение: Аксиомы ФИП:

- аксиомы формализованного исчисления высказываний:

$$\begin{aligned} &(\mathsf{A1}): \quad F \to (G \to F) \\ &(\mathsf{A2}): \quad \big(F \to (G \to F)\big) \to \big((F \to G) \to (F \to H)\big) \\ &(\mathsf{A3}): \quad (\neg G \to \neg F) \to \big((\neg G \to F) \to G\big) \end{aligned}$$

  F, G, H - любые формулы исчисления предикатов.

- предикатные аксиомы (схемы аксиом), т.е. аксиомы с кванторами - аксиомы Бернайса:

$$\begin{aligned} &(\mathsf{PA1}): \quad (\forall x)(F(x) \to F(y)) \\ &(\mathsf{PA2}): \quad F(y) \to (\exists x)(F(x)) \end{aligned}$$

  F(x) — любая формула, содержащая свободные вхождения x, причем ни одно из них не находится в области действия квантора по y (если таковой имеется)

Определение: Правила вывода ФИП:

$$\text{правило отсечения}(\textit{modus ponens}) : \frac{F, F \to G}{G}$$

$$\forall - \text{правило (обобщения)} : \frac{F \to G(x)}{F \to (\forall x)(G(x))}$$

$$\exists - \text{правило (конкретизации)} : \frac{G(x) \to F}{(\exists x)(G(x)) \to F}$$

29 Свойства ФИП

Факт: Полнота, непротиворечивость, разрешимость.
Теорема: Формализованное исчисление предикатов непротиворечиво: одновременно не выводимы две противовоположные формулы ФИП.
Теорема: О разрешимости: формализованное исчисление предикатов неразрешимо
Теорема: о полноте: в ФИП всякая теорема выводима.

30 Понятие формальной аксиоматической теории

Определение: Формальная аксиоматическая теория ФАТ $= \{A, C, P\}$

- задан алфавит теории А - А
- задан алгоритм, который для каждой формулы может проверить корректность записи, синтаксис - С

- задан алгоритм доказательства всех теорем, семантическое правило - P

Истина - все те теоремы что не противоречат исходным теориям. Формализованное исчисление высказываний может служить примером формальной аксиоматической теории.
Определение: Вывод в формальной аксиоматической теории - это всякая последовательность $B_1 \dots B_n$ формул этой теории, такая, что B_i есть либо аксиома теории, либо непосредственное следствие каких-либо предыдущих формул по одному из правил вывода. Формула F теории называется теоремой, если существует вывод в T, последней формулой которого является F.

31 Формальная арифметика (ФА) Пеано

Вторая трактовка: Система аксиом Пеано формальной арифметики состоит из 7 индивидуальных аксиом и одной бесконечной серии аксиом (эту серию аксиом называют схемой индукции):

- $\forall x(\neg(0 = S_x))$
- $\forall x\forall(S_x = S_y \rightarrow x = y)$
- $\forall x(\neg(x = 0) \rightarrow \exists y(x = S_y))$
- $\forall x(x + 0 = x)$
- $\forall x\forall y(x + S_y = S(x + y))$
- $\forall x(x \cdot 0 = 0)$
- $\forall x\forall y(x \cdot (S_y) = (x \cdot y) + x)$
- $(\phi(0) \wedge \forall x(\phi(x) \rightarrow \phi(S_x))) \rightarrow \forall x\phi(x)$,

где $\phi(x)$ произвольная формула со свободной переменной x.

32 Математическая индукция

Определение: Метод математической индукции — специальный метод доказательства, применяемый для доказательства истинности утверждений типа $(\forall x \in \mathbb{N})(P(x))$, то есть $(\forall x)(x \in \mathbb{N} \rightarrow P(x))$. Такие утверждения выражают тот факт, что некоторое свойство P присуще каждому натуральному числу.
Аксиома: Если свойством P обладает число 1 и для всякого натурального числа из того, что оно обладает этим свойством, следует, что и непосредственно следующее за ним натуральное число также обладает им, то и всякое натуральное число обладает свойством P:

$$\Big(P(1) \wedge (\forall x)\big(P(x) \rightarrow P(x+1)\big)\Big) \rightarrow (\forall y)\big(P(y)\big)$$

Факт: Схема доказательства методом математической индукции может быть представлена следующим образом:

- (1): P(1) — устанавливается проверкой
- (2): $(\forall x)\big(P(x) \to P(x+1)\big)$ — доказывается
- (3): $P(1) \wedge (\forall x)\big(P(x) \to P(x+1)\big)$ - из (1), (2) по правилу введения конъюнкции;
- (4): $\big(P(1) \wedge (\forall x)\big(P(x) \to P(x+1)\big)\big) \to (\forall y)\big(P(y)\big)$ — аксиома индукции
- (5): $(\forall y)\big(P(y)\big)$ — из (3), (4) по правилу modus ponens.

P(1) - базой индукции, предположение об истинности утверждения P(x) — предположение индукции, доказательство истинности утверждения P(x+1) — шаг индукции.

33 Элиминация кванторов

Определение: Элиминация кванторов: для формулы существовует такая бескванторная формула B, что в данной теории доказуема (выводима) эквивалентность $A \leftrightarrow B$.
Алгоритм:

- привести в предварённую нормальную форму
- заменить все переменные предикатов всеобщности $\forall$ на x_k переменные множества
- заменить все переменные предикатов существования $\exists$ на n-местные функциональные символы f_k^n

Пример:

1. $\forall a \forall b \forall f \exists C[(O(a) \wedge O(b) \wedge N(f,a,b) \wedge D(f,a,b)) \to (P(C,a,b) \wedge G(f,a,b,C))]$
2. $\forall b \forall f \exists C[(O(x_1) \wedge O(b) \wedge N(f,x_1,b) \wedge D(f,x_1,b)) \to (P(C,x_1,b) \wedge G(f,x_1,b,C))]$
3. $\forall f \exists C[(O(x_1) \wedge O(x_2) \wedge N(f,x_1,x_2) \wedge D(f,x_1,x_2)) \to (P(C,x_1,x_2) \wedge G(f,x_1,x_2,C))]$
4. $\exists C[(O(x_1) \wedge O(x_2) \wedge N(x_3,x_1,x_2) \wedge D(x_3,x_1,x_2)) \to (P(C,x_1,x_2) \wedge G(x_3,x_1,x_2,\mathrm{C}))]$
5. $(O(x_1) \wedge O(x_2) \wedge N(x_3,x_1,x_2) \wedge D(x_3,x_1,x_2)) \to (P(f_1^3,x_1,x_2) \wedge G(x_3,x_1,x_2,f_1^3))$

34 Понятие гёделевской нумерации

Определение: Пусть K — теория первого порядка, содержащая переменные $x_1, x_2 \ldots$, предметные константы $a_1, a_2 \ldots$ функциональные символы f_k^n и предикатные символы A_k^n, где k — номер, а n — арность функционального или предикатного символа.
Способ обозначения объектов ФА, например g(u):

- $G(() = 3$
- $G()) = 5$
- $G(,) = 7$
- $G(\neg) = 9$
- $G(\rightarrow) = 11$
- $G(x_k) = 5 + 8k,\ k = 1, 2, \ldots$
- $G(a_k) = 7 + 8k,\ k = 1, 2, \ldots$
- $G(f_k^n) = 9 + 8 \cdot 2^n 3^k,\ k, n \geqslant 1;$
- $G(A_k^n) = 11 + 8 \cdot 2^n 3^k,\ k, n \geqslant 1.$

Гёделев номер произвольной последовательности $e_0, \ldots, e_r$ выражений определим следующим образом:

$$G(e_0 \ldots e_r) = 2^{G(e_0)} \cdot 3^{G(e_1)} \cdot \ldots \cdot p_r^{G(e_r)}$$

Пример:

$$G(A_1^2(x_1, x_2)) = 2^{G(A_1^2)} \cdot 3^{G(()} \cdot 5^{G(x_1)} \cdot 7^{G(,)} \cdot 11^{G(x_2)} \cdot 13^{G())} = 2^{107} \cdot 3^3 \cdot 5^{13} \cdot 7^7 \cdot 11^{21} \cdot 13^5$$

35 Первая теорема Гёделя о неполноте ФА

Теорема: Гёделя о неполноте (первая):

$$\exists \Phi : \vDash \Phi \Rightarrow \nvdash \Phi$$

Если ФА непротиворечива, то она не полна.
Пример:

$$\Phi \doteqdot \text{“\textit{не существует доказательства формулы} } \Phi\text{”}$$
$$\Phi \doteqdot \left(\neg \exists G \left[Prf(\Phi)\right] = n\right)$$

36 Вторая теорема Гёделя о неполноте ФА

Теорема: Гёделя о неполноте (вторая):

$$\exists \varPhi : \nvdash \varPhi \ \Leftrightarrow \vDash \varPhi$$

Если ткорема не выводима, то она истина, и наоборот.

37 Теорема Чёрча о неразрешимости ФА

Теорема Не существует общего алгоритма, позволяющего за конечное число шагов определить, является ли заданная формула формальной арифметики доказуемой, или нет.

38 Теорема Тарского о понятии истинности в ФА

Теорема: Не всё что истинно является доказуемым.
Пример: Точка - объект, не имещий никаких свойств в математике.

39 Машина Тьюринга (МТ)

Определение: Машина Тьюринга - не физическая машина, а математический объект, как функция, производная..., моделирующая реальные процессы. Работатет с лентой ячеек.
Обладает:

- внешним алфавитом $A = \{a_0, a_1 \dots a_n\}$ - все доступные символы. a_0 - пустой символ
- множеством состояний $Q = \{q_0, q_1 \dots q_m\}$, q_1 - начало работы, q_0 - остановка.
- программой(функциональной схемой), которая определяет работу машины и состоит из команд вида $K(i,j)\colon q_i a_j \to q_k a_l X$, где $X \in \{$С,П,Л$\}$

Суть команды: имея состояние q_i, машина стирает значение a_j обозреваемой ячейки, записывает символ a_l, переходит в новое состояние q_k и перемещается в другую ячейку (или нынешнюю), согласно одному из указаний $\{$С - стоять,П - правая,Л - левая$\}$
Только одна команда соответствует $q_i a_j$, следовательно программа распологает $m(n+1)$ командами.

40 Представление функциональной схемы МТ в табличном виде

Факт: Имея алфавит $A = \{a_0, a_1, a_2\}$, внутренние состояния $Q = \{q_0, q_1, q_2, q_3\}$ и программу $q_1a_1 \to q_1a_1$Л, $q_2a_1 \to q_3a_1$П, $q_3a_1 \to q_1a_1$Л, $q_1a_1 \to q_2a_1$Л, $q_2a_1 \to q_2a_1$Л, $q_3a_1 \to q_3a_1$П, $q_1a_0 \to q_0a_1$, $q_2a_0 \to q_2a_0$Л, $q_3a_0 \to q_3a_0$П, для работы с машиной удобно составить таблицу:

$A\|Q$	q_1	q_2	q_3
a_0	q_0a_1	q_2a_0Л	q_3a_0П
a_1	q_1a_1Л	q_1a_2П	q_1a_1Л
a_2	q_2a_1Л	q_2a_2Л	q_3a_2П

Команды записываются в ячейки на перечении q_ia_j, так как пара q_ia_j однозначно их определяет.

41 Понятие функции вычислимой по Тьюрингу

Определение: Функция вычислима по Тьюрингу, если существует машина Тьюринга, которая вычисляет ее значения, если функция определена для данных аргументов, которая работает вечно, если функция не определена.
Пример: $f(x) = \frac{x}{10}$ - определена только для чисел кратных 10ти:

$$\text{МТ}(x) = \begin{cases} \text{оставить на ленте} \frac{x}{10}\text{, если x кратно десяти} \\ \text{работать бесконечно, если x не кратно десяти} \end{cases}$$

42 Простейшие функции вычислимые по Тьюрингу

Примеры:

- $I_m^n(x_1, x_2, \ldots, x_n) = x_m$ - функции-проекторы
- $S(x) = x + 1$ - добавление единицы
- $O(x) = 0$ нуль-функция
- $K(x, y) = x + y$ функция-сложения
- $J(x, y) = x * y$ функция-умножения

Имея только эти функции, при помощи композиции, уже можно получить множество других функций.

43 Тезис Тьюринга

Определение: Тезис(основная гипотеза теории алгоритмов) Тьюринга: для нахождения значений функции, заданной в некотором алфавите, тогда и только тогда существует какой-нибудь алгоритм, когда функция является вычислимой по Тьюрингу.

44 Композиция нескольких МТ

Определение: Композиция (произведение) машины Θ_1 на машину Θ_2 - новая машина Θ_3 с тем же внешним алфавитом $\{a_0, a_1 \dots a_m\}$, состояниями $\{q_0, q_1 \dots q_n, q_{n+1}, \dots, q_{n+t}\}$ и программой, которая составлена заменой во всех командах первой машины состояния q_0 на состояние q_{n+1} второй машины.
Таким образом, композицию можно получить просто продолжив работу одной машины работой другой.

45 Понятие функции вычислимой по Чёрчу

Класс алгоритмически вычислимых частичных функций совпадает с классом всех частично рекурсивных функций.
Факт: Теория рекурсивных функций включает следующие классы функций: класс примитивных рекурсивных функций, класс общерекурсивных функции и класс частично рекурсивных функций. Теория рекурсивных функций:
1) Задается базис элементарных функций:

- $S(x) = x + 1$ функция следования
- $O(x) = 0$ нуль-функция (константа)
- $I_m^n(x_1, x_2 \dots x_n) = x_m$ функция-проектор

2) Определяются специальные операции над функциями:

- оператор суперпозиции

$$\varphi(x_1 \dots x_n) = f\big(g_1(x_1 \dots x_n),\, \dots,\, g_m(x_1 \dots x_n)\big)$$

- оператор примитивной рекурсии

$$\varphi(x_1 \dots x_n, 0) = f(x_1 \dots x_n);$$
$$\varphi(x_1 \dots x_n, y+1) = g\big(x_1 \dots x_n, y, \varphi(x_1 \dots x_n, y)\big).$$

- оператор минимизации

$$\begin{aligned}
&f_1(x_1 \dots x_n, 0) \neq f_2(x_1 \dots x_n, 0) \\
&\dots \\
&f_1(x_1 \dots x_n, y-1) \neq f_2(x_1 \dots x_n, y-1) \\
&f_1(x_1 \dots x_n, y) = f_2(x_1 \dots x_n, y) \\
&\varphi(x_1 \dots x_n) = \text{ряд}\big[f_1(x_1 \dots x_n, y) = f_2(x_1 \dots x_n, y)\big]
\end{aligned}$$

3) В результате применения определенного количества операций к элементарным функциям, строятся другие

46 Простейшие примитивно рекурсивные функции

Определение: Функция примитивно рекурсивна, если она может быть получена из исходных простейших функций (нуль, следование, проектор) с помощью конечного числа применений операторов суперпозиции и примитивной рекурсии.

47 Понятие общерекурсивной функции

Определение: Общерекурсивные функции — это подмножество частично рекурсивных функций, определённых для всех значений аргументов.
Факт: Задача определения того, является ли частично рекурсивная функция с данным описанием общерекурсивной или нет, алгоритмически неразрешима.

48 Понятие частично рекурсивной функции

Определение: Функция частично рекурсивна, если она может быть получена из простейших функций O, S, I^n_m с помощью конечного числа применений суперпозиции, примитивной рекурсии и оператора минимизации. Как было показано Гёделем, частично рекурсивные функции совпадают с множеством вычислимых функций.

49 Тезис Чёрча

Определение: Тезиса Чёрча: Числовая функция тогда и только тогда алгоритмически (или машинно) вычислима, когда она частично рекурсивна. Класс алгоритмически вычислимых частичных функций совпадает с классом всех частично рекурсивных функций.

50 Теорема об эквивалентности множества функций вычислимых по Тьюрингу и множества функций вычислимых по Чёрчу

Теорема: Функция вычислима по Тьюрингу тогда и только тогда, когда она вычислима по Чёрчу.
Теорема: Класс всех частично рекурсивных функций совпадает с классом всех функций, вычислимых по Тьюрингу.
Теорема: Множества функций вычислимых по Тьюрингу и множества функций вычислимых по Чёрчу эквивалентны

51 Алгоритм как словарная функция

Так как любая задача и результат её выполнения могут быть представлены в виде слова в некотором алфавите, то алгоритм можно определить как словарная функция:

$$A : B \to C$$

Алгоритм как функция отображает входные данные из алфавита B в алфавит C.

52 Свойства алгоритма

Факт:

1. Наличие входных и выходных данных. Данные должны быть допустимыми, алгоритм может выдать ответ за конечное время и количество операций, зациклиться или закончится ошибкой - всё из-за входных
2. Дискретность – есть чёткая последовательность операций, выполняя которую, мы получаем конкретный результат
3. Детерминированность - один и тот же ответ для всех пользователей(результат работы алгоритма зависит только от входных данных)
4. Массовый характер – может быть применен к широкому множеству больших данных (позволяет решить целый класс однотипных задач)

53 Теорема Райса

Теорема: Если заданн класс C функции, нельзя понять, что делает программа не анализируя её. Пусть C — любой непустой класс вычислимых функций, тогда не существует алгоритма, который бы по номеру x функции f_x определял бы, принадлежит f_x классу C или нет. Иначе говоря, множество $\{x\colon\ f_x \in C\}$ неразрешимо.

54 Алгоритмически неразрешимые массовые проблемы

Определение: Алгоритмически неразрешимая задача - задача, для которой не существует алгоритма, который позволяет единым методом решить любую задачу из класса. Неразрешимость какой-либо массовой задачи как правило доказывается путем ее сведения к другой задаче, про которую уже известно, что она является неразрешимой.
Пример: Проблемы определения самоприменимости и проблема остановки.

- **Теорема:** О неразрешимости проблемы остановки: Не существует алгоритма, который для заданного алгоритма A и слова w проверяет, что значение $A(w)$ определено - остановится ли машина или нет.
- **Теорема:** О неразрешимости определения самоприменимости алгоритмов - не существует алгоритма, который по заданному определяет, является ли алгоритм А самоприменимым или нет. Самоприменимый алгоритм, например, для машины Тьюринга - это возможностью обрабатывать свой собственный код. $T(\langle T \rangle) = V$

Другие примеры:

- Проблема соответствий Поста
- Выполнимость оператора
- Проблема равенства слов

55 Временна́я функция сложности алгоритма

Определение: Временна́я функция сложности алгоритма - такая функция $C(r)$ от объёма входных данных r, что показывает количество времени, которое необходимо для завершения работы алгоритма на входных данных. В данном случае, время абстрактно (не секунды, минуты - шаг).

56 Сравнение функций сложности на основе их асимптотических оценок

Факт: Если $C(r)$ - временна́я функция сложности от объёма входных данных r и

$$\lim_{r \to \infty} \frac{C_1(r)}{C_2(r)} = X$$

- $X = \infty \;\Rightarrow\; C_1(r) = \Omega\big(C_2(r)\big)$
- $X = const \;\Rightarrow\; C_1(r) = O\big(C_2(r)\big)$
- $X = 1 \;\Rightarrow\; C_1(r) \equiv C_2(r)$ (не равны, а асимптотически эквивалентны)

57 Шкала асимптотической сложности алгоритмов

$$\ln\ln r \overset{\text{ещё легче}}{\cdots} \ln r \;\cdots < \cdots\; \overset{\text{полином}}{\frac{1}{r} \cdots r^k} \;\cdots < \cdots\; e^r \overset{\text{сложно}}{\cdots} r! \cdots r^r$$

58 Классификация массовых проблем по сложности

Факт: Массовые проблемы: алгоритмически разрешимые, алгоритмически неразрешимые.

	I *класс*	
	II *класс*	P
exp(r)	III *класс*	
	IV *класс*	NP

I $\text{класс} : C(r) \leq k * r$
II $\text{класс} : C(r) \leq p_k(r)$ - полином
III $\text{класс} : C(r) \leq e^r$
IV $\text{класс} : C(r) \geq e^r$

59 Понятие P-разрешимой массовой проблемы

Определение: Проблема P - это проблема решения с помощью полиномиального алгоритма, которая может быть решена за полиномиальное время.

60 Труднорешаемые массовые проблемы

Определение: NP(недетерминированный полином) - это класс труднорешаемых проблем, решение которых можно проверить за полиномиальное время. Так как задачу из этого класса можно представить как набор бесконечных детерминированных P-задач, то решить её за полиномиальное время можно, если отгадать результат и проверить.
Класс $P \subseteq NP$ – это множество задач, для которых существуют полиномиальные алгоритмы решения.
Определение: NP-Complete - пока ни один алгоритм с полиномиальным временем не может ее решить, результат отрицательный (она не решена).