Введение в логику предикатов

Математическая логика и теория алгоритмов

Алексей Романов

19 сентября 2024 г.

МИЭТ

Структура атомарных высказываний

- Вспомним примеры простых высказываний с прошлой лекции: $4 < 5$, «Волга впадает в Балтийское море».

Структура атомарных высказываний

- Вспомним примеры простых высказываний с прошлой лекции: $4 < 5$, «Волга впадает в Балтийское море».
- 4, 5, «Волга» и «Балтийское море» здесь *константы*, обозначающие какие-то объекты.
- $<$ и «_ впадает в _» — *предикаты*, обозначающие отношения между объектами. Оба предиката *бинарные* (или *двухместные*).

Структура атомарных высказываний

- Вспомним примеры простых высказываний с прошлой лекции: $4 < 5$, «Волга впадает в Балтийское море».
- 4, 5, «Волга» и «Балтийское море» здесь *константы*, обозначающие какие-то объекты.
- $<$ и «_ впадает в _» — *предикаты*, обозначающие отношения между объектами. Оба предиката *бинарные* (или *двухместные*).
- Свойства (например «_ круглое» в «Земля круглая») — тоже предикаты, но *унарные*.

Структура атомарных высказываний

- Вспомним примеры простых высказываний с прошлой лекции: $4 < 5$, «Волга впадает в Балтийское море».
- 4, 5, «Волга» и «Балтийское море» здесь *константы*, обозначающие какие-то объекты.
- $<$ и «_ впадает в _» — *предикаты*, обозначающие отношения между объектами. Оба предиката *бинарные* (или *двухместные*).
- Свойства (например «_ круглое» в «Земля круглая») — тоже предикаты, но *унарные*.
- Ещё пример: $2 + 2 = 5$. Здесь 2 и 5 —

Структура атомарных высказываний

- Вспомним примеры простых высказываний с прошлой лекции: $4 < 5$, «Волга впадает в Балтийское море».
- 4, 5, «Волга» и «Балтийское море» здесь *константы*, обозначающие какие-то объекты.
- $<$ и «_ впадает в _» — *предикаты*, обозначающие отношения между объектами. Оба предиката *бинарные* (или *двухместные*).
- Свойства (например «_ круглое» в «Земля круглая») — тоже предикаты, но *унарные*.
- Ещё пример: $2 + 2 = 5$. Здесь 2 и 5 — константы, $=$ —

Структура атомарных высказываний

- Вспомним примеры простых высказываний с прошлой лекции: $4 < 5$, «Волга впадает в Балтийское море».
- 4, 5, «Волга» и «Балтийское море» здесь *константы*, обозначающие какие-то объекты.
- $<$ и «_ впадает в _» — *предикаты*, обозначающие отношения между объектами. Оба предиката *бинарные* (или *двухместные*).
- Свойства (например «_ круглое» в «Земля круглая») — тоже предикаты, но *унарные*.
- Ещё пример: $2 + 2 = 5$. Здесь 2 и 5 — константы, $=$ — предикат, а $+$ —

Структура атомарных высказываний

- Вспомним примеры простых высказываний с прошлой лекции: $4 < 5$, «Волга впадает в Балтийское море».
- 4, 5, «Волга» и «Балтийское море» здесь *константы*, обозначающие какие-то объекты.
- $<$ и «_ впадает в _» — *предикаты*, обозначающие отношения между объектами. Оба предиката *бинарные* (или *двухместные*).
- Свойства (например «_ круглое» в «Земля круглая») — тоже предикаты, но *унарные*.
- Ещё пример: $2 + 2 = 5$. Здесь 2 и 5 — константы, $=$ — предикат, а $+$ — *функция*.
- $2 + 2$ — *терм*, обозначает объект (как и константы).
- В n-местный предикат можно подставить n термов и получить высказывание.

Сигнатуры, термы, формулы

- *Сигнатура* Σ — конечные или счётные множества константных $Const_\Sigma$, функциональных Fun_Σ и предикатных $Pred_\Sigma$ символов. Для каждого символа из Fun_Σ и $Pred_\Sigma$ задано число аргументов (*арность*).
- *Переменные* —

Сигнатуры, термы, формулы

- *Сигнатура* Σ — конечные или счётные множества константных $Const_{\Sigma}$, функциональных Fun_{Σ} и предикатных $Pred_{\Sigma}$ символов. Для каждого символа из Fun_{Σ} и $Pred_{\Sigma}$ задано число аргументов (*арность*).
- *Переменные* — $x, y, z_3, \ldots$. Обозначают

Сигнатуры, термы, формулы

- *Сигнатура* Σ — конечные или счётные множества константных $Const_\Sigma$, функциональных Fun_Σ и предикатных $Pred_\Sigma$ символов. Для каждого символа из Fun_Σ и $Pred_\Sigma$ задано число аргументов (*арность*).
- *Переменные* — $x, y, z_3, \ldots$. Обозначают какие-то объекты (не истину/ложь, как p, q, r). Множество Var не зависит от сигнатуры.

Сигнатуры, термы, формулы

- *Сигнатура* Σ — конечные или счётные множества константных $Const_\Sigma$, функциональных Fun_Σ и предикатных $Pred_\Sigma$ символов. Для каждого символа из Fun_Σ и $Pred_\Sigma$ задано число аргументов (*арность*).
- *Переменные* — x, y, z_3, $\ldots$. Обозначают какие-то объекты (не истину/ложь, как p, q, r). Множество Var не зависит от сигнатуры.
- *Термы* —

Сигнатуры, термы, формулы

- *Сигнатура* Σ — конечные или счётные множества константных $Const_\Sigma$, функциональных Fun_Σ и предикатных $Pred_\Sigma$ символов. Для каждого символа из Fun_Σ и $Pred_\Sigma$ задано число аргументов (*арность*).
- *Переменные* — x, y, z_3, $\ldots$. Обозначают какие-то объекты (не истину/ложь, как p, q, r). Множество Var не зависит от сигнатуры.
- *Термы* — x, $(y + 2) \cdot z$, $\ldots$ Выражения, значения которых — объекты. Строятся из переменных и константных символов применением функциональных. Множество термов над Σ: $Term_\Sigma$.

Сигнатуры, термы, формулы

- *Сигнатура* Σ — конечные или счётные множества константных $Const_\Sigma$, функциональных Fun_Σ и предикатных $Pred_\Sigma$ символов. Для каждого символа из Fun_Σ и $Pred_\Sigma$ задано число аргументов (*арность*).
- *Переменные* — $x, y, z_3, \dots$. Обозначают какие-то объекты (не истину/ложь, как p, q, r). Множество Var не зависит от сигнатуры.
- *Термы* — $x, (y + 2) \cdot z, \dots$ Выражения, значения которых — объекты. Строятся из переменных и константных символов применением функциональных. Множество термов над Σ: $Term_\Sigma$.
- *Формулы* —

Сигнатуры, термы, формулы

- *Сигнатура* Σ — конечные или счётные множества константных $Const_\Sigma$, функциональных Fun_Σ и предикатных $Pred_\Sigma$ символов. Для каждого символа из Fun_Σ и $Pred_\Sigma$ задано число аргументов (*арность*).
- *Переменные* — $x, y, z_3, \ldots$. Обозначают какие-то объекты (не истину/ложь, как p, q, r). Множество Var не зависит от сигнатуры.
- *Термы* — $x, (y + 2) \cdot z, \ldots$ Выражения, значения которых — объекты. Строятся из переменных и константных символов применением функциональных. Множество термов над Σ: $Term_\Sigma$.
- *Формулы* — $x = y + 1$; $\forall x\ x \neq x^2\ \ldots$ Вот их значения истина и ложь. *Атомарные формулы* строятся из термов применением предикатных символов, а остальные применением связок $\wedge/\vee/\ldots$ и кванторов $\forall$ и $\exists$ к формулам. Множество формул над Σ: $Form_\Sigma$.

Модели

- *Модель* M сигнатуры Σ состоит из:
    - *Носитель* (или *универсум*):

Модели

- *Модель* M сигнатуры Σ состоит из:
    - *Носитель* (или *универсум*): множество $\bar{M}$.
    - Для каждого символа $c : Const_\Sigma$[1]:

[1] Я пишу $:$ вместо $\in$ для объявления новой переменной.

Модели

- *Модель* M сигнатуры Σ состоит из:
  - *Носитель* (или *универсум*): множество $\bar{M}$.
  - Для каждого символа $c : Const_\Sigma$[1]: элемент $c_M : \bar{M}$.
  - Для $f : Fun_\Sigma$:

[1] Я пишу $:$ вместо $\in$ для объявления новой переменной.

Модели

- *Модель* M сигнатуры Σ состоит из:
  - *Носитель* (или *универсум*): множество $\bar{M}$.
  - Для каждого символа $c : Const_\Sigma$[1]: элемент $c_M : \bar{M}$.
  - Для $f : Fun_\Sigma$: функция $f_M : M^{arity(f)} \to M$.
  - Для $P : Pred_\Sigma$:

[1] Я пишу $:$ вместо $\in$ для объявления новой переменной.

Модели

- *Модель* M сигнатуры Σ состоит из:
    - *Носитель* (или *универсум*): множество $\bar{M}$.
    - Для каждого символа $c : Const_\Sigma$[1]: элемент $c_M : \bar{M}$.
    - Для $f : Fun_\Sigma$: функция $f_M : M^{arity(f)} \to M$.
    - Для $P : Pred_\Sigma$: предикат $P_M : M^{arity(P)} \to \{0, 1\}$.
- *Оценка* — значение переменных в модели, $\sigma : Var \to \bar{M}$. Или $V \to \bar{M}$, где $V \subset Var$.
- Важно: символы сигнатуры сами по себе ничего не говорят об их интерпретации! $+_M$ может быть $-$ или $\cdot$ или $x, y \mapsto \int_x^y t dt$.

[1] Я пишу $:$ вместо $\in$ для объявления новой переменной.

Модели

- *Модель* M сигнатуры Σ состоит из:
  - *Носитель* (или *универсум*): множество $\bar{M}$.
  - Для каждого символа $c : Const_\Sigma$[1]: элемент $c_M : \bar{M}$.
  - Для $f : Fun_\Sigma$: функция $f_M : M^{arity(f)} \to M$.
  - Для $P : Pred_\Sigma$: предикат $P_M : M^{arity(P)} \to \{0, 1\}$.
- *Оценка* — значение переменных в модели, $\sigma : Var \to \bar{M}$. Или $V \to \bar{M}$, где $V \subset Var$.
- Важно: символы сигнатуры сами по себе ничего не говорят об их интерпретации! $+_M$ может быть $-$ или $\cdot$ или $x, y \mapsto \int_x^y t dt$.
- Одно исключение: если есть $=$, то $=_M$ это равенство на $\bar{M}$.

---

[1] Я пишу $:$ вместо $\in$ для объявления новой переменной.

Значения термов и формул

- σ расширяется на $Term_\Sigma \to$??? и на $Form_\Sigma \to$??? по индукции:

Значения термов и формул

- σ расширяется на $Term_\Sigma \to \bar{M}$ и на $Form_\Sigma \to \{0, 1\}$ по индукции:
- $\sigma(c) = c_M$.
- $\sigma(f(t_1, \ldots, t_n)) = f_M(\sigma(t_1), \ldots, \sigma(t_n))$.

Значения термов и формул

- σ расширяется на $Term_\Sigma \to \bar{M}$ и на $Form_\Sigma \to \{0, 1\}$ по индукции:
- $\sigma(c) = c_M$.
- $\sigma(f(t_1, \ldots, t_n)) = f_M(\sigma(t_1), \ldots, \sigma(t_n))$.

- $\sigma(P(t_1, \ldots, t_n)) = P_M(\sigma(t_1), \ldots, \sigma(t_n))$.
- $\sigma(A \wedge B) = \sigma(A) \wedge \sigma(B)$ (аналогично для $\neg$/$\vee$/$\to$).

Значения термов и формул

- σ расширяется на $Term_\Sigma \to \bar{M}$ и на $Form_\Sigma \to \{0,1\}$ по индукции:
- $\sigma(c) = c_M$.
- $\sigma(f(t_1,\dots,t_n)) = f_M(\sigma(t_1),\dots,\sigma(t_n))$.

- $\sigma(P(t_1,\dots,t_n)) = P_M(\sigma(t_1),\dots,\sigma(t_n))$.
- $\sigma(A \wedge B) = \sigma(A) \wedge \sigma(B)$ (аналогично для $\neg$/$\vee$/$\to$).
- Для кванторов нужно сначала определить $\sigma_{v\mapsto a}$ (где v переменная, а a — объект из $\bar{M}$):

$$\sigma_{v\mapsto a}(v) = a$$
$$\sigma_{v\mapsto a}(u) = \sigma(u) \text{ для остальных переменных } u.$$

Значения термов и формул

- σ расширяется на $Term_\Sigma \to \bar{M}$ и на $Form_\Sigma \to \{0, 1\}$ по индукции:
- $\sigma(c) = c_M$.
- $\sigma(f(t_1, \ldots, t_n)) = f_M(\sigma(t_1), \ldots, \sigma(t_n))$.

- $\sigma(P(t_1, \ldots, t_n)) = P_M(\sigma(t_1), \ldots, \sigma(t_n))$.
- $\sigma(A \wedge B) = \sigma(A) \wedge \sigma(B)$ (аналогично для $\neg$/$\vee$/$\to$).
- Для кванторов нужно сначала определить $\sigma_{v \mapsto a}$ (где v переменная, а a — объект из $\bar{M}$):

  $$\sigma_{v \mapsto a}(v) = a$$
  $$\sigma_{v \mapsto a}(u) = \sigma(u) \text{ для остальных переменных } u.$$

  Теперь

  $$\sigma(\forall v\ A) = \begin{cases} 1, \text{ если } \forall a : \bar{M}\ \sigma_{v \mapsto a}(A) = 1 \\ 0 \text{ иначе} \end{cases}$$

  и аналогично для $\exists$.

Свободные и связанные переменные

- *Связанное вхождение переменной* —

Свободные и связанные переменные

- *Связанное вхождение переменной* — v под квантором по этой же переменной $\forall v / \exists v$.
- *Свободное вхождение переменной* —

Свободные и связанные переменные

- *Связанное вхождение переменной* — v под квантором по этой же переменной $\forall v / \exists v$.
- *Свободное вхождение переменной* — любое другое.
- Переменная свободна в формуле, если имеет какие-то свободные вхождения.
- *Замкнутая формула* (или *предложение*) — формула без свободных переменных.
- Почему это важно? Какая между ними разница?

Свободные и связанные переменные

- *Связанное вхождение переменной* — v под квантором по этой же переменной $\forall v / \exists v$.
- *Свободное вхождение переменной* — любое другое.
- Переменная свободна в формуле, если имеет какие-то свободные вхождения.
- *Замкнутая формула* (или *предложение*) — формула без свободных переменных.
- Почему это важно? Какая между ними разница?
- $\sigma(\forall / \exists v \ \ldots)$ не зависит от $\sigma(v)$.

Свободные и связанные переменные

- *Связанное вхождение переменной* — v под квантором по этой же переменной $\forall v / \exists v$.
- *Свободное вхождение переменной* — любое другое.
- Переменная свободна в формуле, если имеет какие-то свободные вхождения.
- *Замкнутая формула* (или *предложение*) — формула без свободных переменных.
- Почему это важно? Какая между ними разница?
- $\sigma(\forall/\exists v\ \ldots)$ не зависит от $\sigma(v)$.
- Отсюда можно доказать, что значение формулы A зависит только от значений свободных переменных A, но не от связанных.

Свободные и связанные переменные

- *Связанное вхождение переменной* — v под квантором по этой же переменной $\forall v / \exists v$.
- *Свободное вхождение переменной* — любое другое.
- Переменная свободна в формуле, если имеет какие-то свободные вхождения.
- *Замкнутая формула* (или *предложение*) — формула без свободных переменных.
- Почему это важно? Какая между ними разница?
- $\sigma(\forall / \exists v\ \dots)$ не зависит от $\sigma(v)$.
- Отсюда можно доказать, что значение формулы A зависит только от значений свободных переменных A, но не от связанных.
- А значение замкнутой формулы?

Свободные и связанные переменные

- *Связанное вхождение переменной* — v под квантором по этой же переменной $\forall v / \exists v$.
- *Свободное вхождение переменной* — любое другое.
- Переменная свободна в формуле, если имеет какие-то свободные вхождения.
- *Замкнутая формула* (или *предложение*) — формула без свободных переменных.
- Почему это важно? Какая между ними разница?
- $\sigma(\forall/\exists v\ \ldots)$ не зависит от $\sigma(v)$.
- Отсюда можно доказать, что значение формулы A зависит только от значений свободных переменных A, но не от связанных.
- А значение замкнутой формулы? Вообще не зависит от оценки, только от модели.

Свободные и связанные переменные (2)

- Связанную переменную можно переименовать, не изменив смысла и значения формулы.
- Вместо свободной переменной можно подставить произвольный терм, вместо связанной нельзя.
- Переменные с одним названием, связанные разными кванторами, это по сути разные переменные.
  Пример: $x = 3 \wedge (\exists x\, x = 0) \wedge \forall x\, (x > 0 \to x^2 > 0)$.
- В математике переменные могут связываться не только кванторами:

Свободные и связанные переменные (2)

- Связанную переменную можно переименовать, не изменив смысла и значения формулы.
- Вместо свободной переменной можно подставить произвольный терм, вместо связанной нельзя.
- Переменные с одним названием, связанные разными кванторами, это по сути разные переменные. Пример: $x = 3 \wedge (\exists x\ x = 0) \wedge \forall x\ (x > 0 \to x^2 > 0)$.
- В математике переменные могут связываться не только кванторами:

$$\lim_{x \to \dots} \dots \qquad \int_{\dots}^{\dots} \dots dx \qquad \{x | \dots\}$$

все связывают x.

Примеры

- Рассмотрим пример.
- Формула (замкнутая): $\forall x \, (P(x) \to Q(x))$.
- Возьмём произвольную модель из 3 элементов: $\bar{M} = \{0, 1, 2\}$.
- Как можно задать предикаты P и Q?

Примеры

- Рассмотрим пример.
- Формула (замкнутая): $\forall x\ (P(x) \to Q(x))$.
- Возьмём произвольную модель из 3 элементов: $\bar{M} = \{0, 1, 2\}$.
- Как можно задать предикаты P и Q?

x	0	1	2
P(x)	0	0	1
Q(x)	1	0	1

- Истинна ли формула?

Примеры

- Рассмотрим пример.
- Формула (замкнутая): $\forall x \ (P(x) \to Q(x))$.
- Возьмём произвольную модель из 3 элементов: $\bar{M} = \{0, 1, 2\}$.
- Как можно задать предикаты P и Q?

x	0	1	2
P(x)	0	0	1
Q(x)	1	0	1

- Истинна ли формула? Да.

Примеры

- Рассмотрим пример.
- Формула (замкнутая): $\forall x\ (P(x) \to Q(x))$.
- Возьмём произвольную модель из 3 элементов: $\bar{M} = \{0, 1, 2\}$.
- Как можно задать предикаты P и Q?

x	0	1	2
P(x)	0	0	1
Q(x)	1	0	1

- Истинна ли формула? Да.
- Попробуем другую формулу на той же модели: $\exists x\ (P(x) \wedge Q(x)) \wedge \exists x\ \neg P(x)$. Истинна ли она?

Примеры

- Рассмотрим пример.
- Формула (замкнутая): $\forall x \; (P(x) \to Q(x))$.
- Возьмём произвольную модель из 3 элементов: $\bar{M} = \{0, 1, 2\}$.
- Как можно задать предикаты P и Q?

x	0	1	2
P(x)	0	0	1
Q(x)	1	0	1

- Истинна ли формула? Да.
- Попробуем другую формулу на той же модели: $\exists x \; (P(x) \wedge Q(x)) \wedge \exists x \; \neg P(x)$. Истинна ли она? Да.

Примеры

- Рассмотрим пример.
- Формула (замкнутая): $\forall x\ (P(x) \to Q(x))$.
- Возьмём произвольную модель из 3 элементов: $\bar{M} = \{0, 1, 2\}$.
- Как можно задать предикаты P и Q?

x	0	1	2
P(x)	0	0	1
Q(x)	1	0	1

- Истинна ли формула? Да.
- Попробуем другую формулу на той же модели: $\exists x\ (P(x) \wedge Q(x)) \wedge \exists x\ \neg P(x)$. Истинна ли она? Да.
- Как насчёт незамкнутой формулы $P(x) \to \forall x\ Q(x)$?

Примеры

- Рассмотрим пример.
- Формула (замкнутая): $\forall x\ (P(x) \rightarrow Q(x))$.
- Возьмём произвольную модель из 3 элементов: $\bar{M} = \{0, 1, 2\}$.
- Как можно задать предикаты P и Q?

x	0	1	2
P(x)	0	0	1
Q(x)	1	0	1

- Истинна ли формула? Да.
- Попробуем другую формулу на той же модели: $\exists x\ (P(x) \wedge Q(x)) \wedge \exists x\ \neg P(x)$. Истинна ли она? Да.
- Как насчёт незамкнутой формулы $P(x) \rightarrow \forall x\ Q(x)$? Истинна при $\sigma(x) = 0,\ \sigma(x) = 1$, ложна при $\sigma(x) = 2$.

Примеры

- Рассмотрим пример.
- Формула (замкнутая): $\forall x\ (P(x) \to Q(x))$.
- Возьмём произвольную модель из 3 элементов: $\bar{M} = \{0, 1, 2\}$.
- Как можно задать предикаты P и Q?

x	0	1	2
P(x)	0	0	1
Q(x)	1	0	1

- Истинна ли формула? Да.
- Попробуем другую формулу на той же модели: $\exists x\ (P(x) \land Q(x)) \land \exists x\ \neg P(x)$. Истинна ли она? Да.
- Как насчёт незамкнутой формулы $P(x) \to \forall x\ Q(x)$? Истинна при $\sigma(x) = 0,\ \sigma(x) = 1$, ложна при $\sigma(x) = 2$.
- Для простоты можно писать $x = 0$ вместо $\sigma(x) = 0$.

Примеры

- Рассмотрим пример.
- Формула (замкнутая): $\forall x\ (P(x) \to Q(x))$.
- Возьмём произвольную модель из 3 элементов: $\bar{M} = \{0, 1, 2\}$.
- Как можно задать предикаты P и Q?

x	0	1	2
P(x)	0	0	1
Q(x)	1	0	1

- Истинна ли формула? Да.
- Попробуем другую формулу на той же модели: $\exists x\ (P(x) \wedge Q(x)) \wedge \exists x\ \neg P(x)$. Истинна ли она? Да.
- Как насчёт незамкнутой формулы $P(x) \to \forall x\ Q(x)$? Истинна при $\sigma(x) = 0,\ \sigma(x) = 1$, ложна при $\sigma(x) = 2$.
- Для простоты можно писать $x = 0$ вместо $\sigma(x) = 0$.
- Ничего не изменится, если $\bar{M} = \{a, b, c\}$, какие-то абстрактные объекты.

Примеры

- Ещё пример с бинарным предикатом:
- Формула: $\forall x \exists y\ (P(x, y) \to P(x, x))$.
- Возьмём $\bar{M} = \{a, b, c\}$.
- Какой таблицей задаётся бинарный предикат?

Примеры

- Ещё пример с бинарным предикатом:
- Формула: $\forall x \exists y \ (P(x, y) \to P(x, x))$.
- Возьмём $\bar{M} = \{a, b, c\}$.
- Какой таблицей задаётся бинарный предикат?

x \ y	a	b	c
a	0	1	1
b	0	1	0
c	1	1	0

- Истинна ли формула?

Примеры

- Ещё пример с бинарным предикатом:
- Формула: $\forall x \exists y\ (P(x, y) \to P(x, x))$.
- Возьмём $\bar{M} = \{a, b, c\}$.
- Какой таблицей задаётся бинарный предикат?

x \ y	a	b	c
a	0	1	1
b	0	1	0
c	1	1	0

- Истинна ли формула? Да.

Примеры

- Ещё пример с бинарным предикатом:
- Формула: $\forall x \exists y\ (P(x, y) \to P(x, x))$.
- Возьмём $\bar{M} = \{a, b, c\}$.
- Какой таблицей задаётся бинарный предикат?

x \ y	a	b	c
a	0	1	1
b	0	1	0
c	1	1	0

- Истинна ли формула? Да.
- Слегка поменяем: $\forall x\ ((\exists y\ P(x, y)) \to P(x, x))$. Эта формула

Примеры

- Ещё пример с бинарным предикатом:
- Формула: $\forall x \exists y\ (P(x,y) \to P(x,x))$.
- Возьмём $\bar{M} = \{a, b, c\}$.
- Какой таблицей задаётся бинарный предикат?

x \ y	a	b	c
a	0	1	1
b	0	1	0
c	1	1	0

- Истинна ли формула? Да.
- Слегка поменяем: $\forall x\ ((\exists y\ P(x,y)) \to P(x,x))$. Эта формула ложна.

Примеры

- Ещё пример с бинарным предикатом:
- Формула: $\forall x \exists y \ (P(x, y) \to P(x, x))$.
- Возьмём $\bar{M} = \{a, b, c\}$.
- Какой таблицей задаётся бинарный предикат?

x \ y	a	b	c
a	0	1	1
b	0	1	0
c	1	1	0

- Истинна ли формула? Да.
- Слегка поменяем: $\forall x \ ((\exists y \ P(x, y)) \to P(x, x))$. Эта формула ложна.
- На бесконечных моделях работает так же, но мы не можем просто перечислить все значения переменных, чтобы найти значение формулы!

Примеры

- Ещё пример с бинарным предикатом:
- Формула: $\forall x \exists y \, (P(x,y) \to P(x,x))$.
- Возьмём $\bar{M} = \{a,b,c\}$.
- Какой таблицей задаётся бинарный предикат?

x \ y	a	b	c
a	0	1	1
b	0	1	0
c	1	1	0

- Истинна ли формула? Да.
- Слегка поменяем: $\forall x \, ((\exists y \, P(x,y)) \to P(x,x))$. Эта формула ложна.
- На бесконечных моделях работает так же, но мы не можем просто перечислить все значения переменных, чтобы найти значение формулы!
- Вместо механического процесса приходится думать.
- На практике для больших конечных тоже.

Формализация

- Часто возникает задача: дано утверждение (математическое или в терминах «реального мира»), нужно записать его в виде формулы данной сигнатуры.
- Пример: «Кто-то любит всех на свете». Универсум: люди, предикат $Loves(x, y)$ для «x любит y».
- Кто-то любит всех на свете $\equiv \exists x$ x любит всех на свете

Формализация

- Часто возникает задача: дано утверждение (математическое или в терминах «реального мира»), нужно записать его в виде формулы данной сигнатуры.
- Пример: «Кто-то любит всех на свете». Универсум: люди, предикат $Loves(x, y)$ для «x любит y».
- Кто-то любит всех на свете $\equiv \exists x$ x любит всех на свете $\equiv \exists x\ \forall y$ x любит y

Формализация

- Часто возникает задача: дано утверждение (математическое или в терминах «реального мира»), нужно записать его в виде формулы данной сигнатуры.
- Пример: «Кто-то любит всех на свете». Универсум: люди, предикат $Loves(x, y)$ для «x любит y».
- Кто-то любит всех на свете $\equiv \exists x$ x любит всех на свете $\equiv \exists x\ \forall y$ x любит y $\equiv \exists x\ \forall y\ Loves(x, y)$.

Формализация

- Часто возникает задача: дано утверждение (математическое или в терминах «реального мира»), нужно записать его в виде формулы данной сигнатуры.
- Пример: «Кто-то любит всех на свете». Универсум: люди, предикат $Loves(x, y)$ для «x любит y».
- Кто-то любит всех на свете $\equiv \exists x$ x любит всех на свете $\equiv \exists x\ \forall y$ x любит y $\equiv \exists x\ \forall y\ Loves(x, y)$.

Ещё примеры формализации

- «x делится на 2». Универсум: натуральные числа.
- Если предположили что-то вроде $x/2 \in \mathbb{N}$: это не подойдёт. Почему?

Ещё примеры формализации

- «x делится на 2». Универсум: натуральные числа.
- Если предположили что-то вроде $x/2 \in \mathbb{N}$: это не подойдёт. Почему?
- Например, $\mathbb{N}$ это не объект нашего универсума. А если $\in \mathbb{N}$ рассматривать как единый предикатный символ, он верен для всех объектов!

Ещё примеры формализации

- «x делится на 2». Универсум: натуральные числа.
- Если предположили что-то вроде $x/2 \in \mathbb{N}$: это не подойдёт. Почему?
- Например, $\mathbb{N}$ это не объект нашего универсума. А если $\in \mathbb{N}$ рассматривать как единый предикатный символ, он верен для всех объектов!
- Более того, если $/$ — функциональный символ, то он должен иметь значение в нашем универсуме для любых аргументов. Есть варианты логики, которые снимают это ограничение, но мы их не изучаем.

Ещё примеры формализации

- «x делится на 2». Универсум: натуральные числа.
- Если предположили что-то вроде $x/2 \in \mathbb{N}$: это не подойдёт. Почему?
- Например, $\mathbb{N}$ это не объект нашего универсума. А если $\in \mathbb{N}$ рассматривать как единый предикатный символ, он верен для всех объектов!
- Более того, если $/$ — функциональный символ, то он должен иметь значение в нашем универсуме для любых аргументов. Есть варианты логики, которые снимают это ограничение, но мы их не изучаем.
- Ответ (возможный):

Ещё примеры формализации

- «x делится на 2». Универсум: натуральные числа.
- Если предположили что-то вроде $x/2 \in \mathbb{N}$: это не подойдёт. Почему?
- Например, $\mathbb{N}$ это не объект нашего универсума. А если $\in \mathbb{N}$ рассматривать как единый предикатный символ, он верен для всех объектов!
- Более того, если $/$ — функциональный символ, то он должен иметь значение в нашем универсуме для любых аргументов. Есть варианты логики, которые снимают это ограничение, но мы их не изучаем.
- Ответ (возможный): $\exists y\ x = 2 \cdot y$.

Ещё примеры формализации

- «x делится на 2». Универсум: натуральные числа.
- Если предположили что-то вроде $x/2 \in \mathbb{N}$: это не подойдёт. Почему?
- Например, $\mathbb{N}$ это не объект нашего универсума. А если $\in \mathbb{N}$ рассматривать как единый предикатный символ, он верен для всех объектов!
- Более того, если $/$ — функциональный символ, то он должен иметь значение в нашем универсуме для любых аргументов. Есть варианты логики, которые снимают это ограничение, но мы их не изучаем.
- Ответ (возможный): $\exists y\ x = 2 \cdot y$.
- Можно ли то же самое записать без $\cdot$?

Ещё примеры формализации

- «x делится на 2». Универсум: натуральные числа.
- Если предположили что-то вроде $x/2 \in \mathbb{N}$: это не подойдёт. Почему?
- Например, $\mathbb{N}$ это не объект нашего универсума. А если $\in \mathbb{N}$ рассматривать как единый предикатный символ, он верен для всех объектов!
- Более того, если $/$ — функциональный символ, то он должен иметь значение в нашем универсуме для любых аргументов. Есть варианты логики, которые снимают это ограничение, но мы их не изучаем.
- Ответ (возможный): $\exists y\ x = 2 \cdot y$.
- Можно ли то же самое записать без $\cdot$? Да! $\exists y\ x = y + y$.

Ещё примеры формализации

- «x делится на 2». Универсум: натуральные числа.
- Если предположили что-то вроде $x/2 \in \mathbb{N}$: это не подойдёт. Почему?
- Например, $\mathbb{N}$ это не объект нашего универсума. А если $\in \mathbb{N}$ рассматривать как единый предикатный символ, он верен для всех объектов!
- Более того, если $/$ — функциональный символ, то он должен иметь значение в нашем универсуме для любых аргументов. Есть варианты логики, которые снимают это ограничение, но мы их не изучаем.
- Ответ (возможный): $\exists y\ x = 2 \cdot y$.
- Можно ли то же самое записать без $\cdot$? Да! $\exists y\ x = y + y$.
- Вот «x делится на y» без $\cdot$ записать уже не получится.

Ещё примеры формализации

- «x делится на 2». Универсум: натуральные числа.
- Если предположили что-то вроде $x/2 \in \mathbb{N}$: это не подойдёт. Почему?
- Например, $\mathbb{N}$ это не объект нашего универсума. А если $\in \mathbb{N}$ рассматривать как единый предикатный символ, он верен для всех объектов!
- Более того, если $/$ — функциональный символ, то он должен иметь значение в нашем универсуме для любых аргументов. Есть варианты логики, которые снимают это ограничение, но мы их не изучаем.
- Ответ (возможный): $\exists y\ x = 2 \cdot y$.
- Можно ли то же самое записать без $\cdot$? Да! $\exists y\ x = y + y$.
- Вот «x делится на y» без $\cdot$ записать уже не получится.

Формализация со свободными переменными

- У нас на предыдущих слайдах появлялись утверждения с переменными (например, «x любит всех на свете») в промежуточных результатах.
- Может и сразу быть дано такое утверждение.

Формализация со свободными переменными

- У нас на предыдущих слайдах появлялись утверждения с переменными (например, «x любит всех на свете») в промежуточных результатах.
- Может и сразу быть дано такое утверждение.
- В результате должна получиться формула с теми же свободными переменными (и какими угодно связанными).

Формализация со свободными переменными

- У нас на предыдущих слайдах появлялись утверждения с переменными (например, «x любит всех на свете») в промежуточных результатах.
- Может и сразу быть дано такое утверждение.
- В результате должна получиться формула с теми же **свободными** переменными (и какими угодно связанными).
- Если в формуле есть «лишние» свободные переменные или связана одна из тех, что есть в формализуемом утверждении, это заведомо неверный ответ.

Многосортная логика предикатов

- Часто удобно одновременно говорить о нескольких разных типах объектов. Пример: числа, множества чисел и функции в мат. анализе. Тогда
- К сигнатуре добавляется набор *сортов*. Каждый сорт обозначает какое-то множество объектов.
- У функциональных и предикатных символов кроме числа аргументов задан сорт каждого, у функциональных ещё и сорт результата.
- Применение символов к аргументам не тех сортов считается бессмысленным (т.е. его результат не является термом/формулой).
- Каждая переменная имеет сорт: $x : S$. Сорт термов определяется по индукции.
- В моделях есть носитель для каждого сорта.
- Многосортную логику можно свести к односортной, добавив по предикату для каждого сорта, но формулы при этом усложняются.

Подстановки

- Попробуем определить результат замены переменной v на терм t в терм s и в формулу A: $s[v \mapsto t]$ и $A[v \mapsto t]$. Должно получиться

$$\sigma(s[v \mapsto t]) = \sigma_{v \mapsto \sigma(t)}(s)$$
$$\sigma(A[v \mapsto t]) = \sigma_{v \mapsto \sigma(t)}(A)$$

(понятен ли смысл этого?).

Подстановки

- Попробуем определить результат замены переменной v на терм t в терм s и в формулу A: $s[v \mapsto t]$ и $A[v \mapsto t]$. Должно получиться

$$\sigma(s[v \mapsto t]) = \sigma_{v \mapsto \sigma(t)}(s)$$
$$\sigma(A[v \mapsto t]) = \sigma_{v \mapsto \sigma(t)}(A)$$

  (понятен ли смысл этого?).
- Первые шаги очевидны:

$$v[v \mapsto t] = t$$

Подстановки

- Попробуем определить результат замены переменной v на терм t в терм s и в формулу A: $s[v \mapsto t]$ и $A[v \mapsto t]$. Должно получиться

$$\sigma(s[v \mapsto t]) = \sigma_{v \mapsto \sigma(t)}(s)$$
$$\sigma(A[v \mapsto t]) = \sigma_{v \mapsto \sigma(t)}(A)$$

  (понятен ли смысл этого?).
- Первые шаги очевидны:

$$v[v \mapsto t] = t$$
$$u[v \mapsto t] =$$

Подстановки

- Попробуем определить результат замены переменной v на терм t в терм s и в формулу A: $s[v \mapsto t]$ и $A[v \mapsto t]$. Должно получиться

$$\sigma(s[v \mapsto t]) = \sigma_{v \mapsto \sigma(t)}(s)$$
$$\sigma(A[v \mapsto t]) = \sigma_{v \mapsto \sigma(t)}(A)$$

  (понятен ли смысл этого?).
- Первые шаги очевидны:

$$v[v \mapsto t] = t$$
$$u[v \mapsto t] = u \text{ (u — переменная, кроме v)}$$
$$c[v \mapsto t] =$$

Подстановки

- Попробуем определить результат замены переменной v на терм t в терм s и в формулу A: $s[v \mapsto t]$ и $A[v \mapsto t]$. Должно получиться

$$\sigma(s[v \mapsto t]) = \sigma_{v \mapsto \sigma(t)}(s)$$
$$\sigma(A[v \mapsto t]) = \sigma_{v \mapsto \sigma(t)}(A)$$

  (понятен ли смысл этого?).
- Первые шаги очевидны:

  $v[v \mapsto t] = t$
  $u[v \mapsto t] = u$ (u — переменная, кроме v)
  $c[v \mapsto t] = c$ (c — константа)

Подстановки

- Попробуем определить результат замены переменной v на терм t в терм s и в формулу A: $s[v \mapsto t]$ и $A[v \mapsto t]$. Должно получиться

$$\sigma(s[v \mapsto t]) = \sigma_{v \mapsto \sigma(t)}(s)$$
$$\sigma(A[v \mapsto t]) = \sigma_{v \mapsto \sigma(t)}(A)$$

  (понятен ли смысл этого?).
- Первые шаги очевидны:

$v[v \mapsto t] = t$
$u[v \mapsto t] = u$ (u — переменная, кроме v)
$c[v \mapsto t] = c$ (c — константа)
$f(t_1, \ldots, t_n)[v \mapsto t] =$

Подстановки

- Попробуем определить результат замены переменной v на терм t в терм s и в формулу A: $s[v \mapsto t]$ и $A[v \mapsto t]$. Должно получиться

$$\sigma(s[v \mapsto t]) = \sigma_{v \mapsto \sigma(t)}(s)$$
$$\sigma(A[v \mapsto t]) = \sigma_{v \mapsto \sigma(t)}(A)$$

  (понятен ли смысл этого?).
- Первые шаги очевидны:

$$v[v \mapsto t] = t$$
$$u[v \mapsto t] = u \text{ (u — переменная, кроме v)}$$
$$c[v \mapsto t] = c \text{ (c — константа)}$$
$$f(t_1, \ldots, t_n)[v \mapsto t] = f(t_1[v \mapsto t], \ldots, t_n[v \mapsto t])$$

Подстановки

- Попробуем определить результат замены переменной v на терм t в терм s и в формулу A: $s[v \mapsto t]$ и $A[v \mapsto t]$. Должно получиться

$$\sigma(s[v \mapsto t]) = \sigma_{v \mapsto \sigma(t)}(s)$$
$$\sigma(A[v \mapsto t]) = \sigma_{v \mapsto \sigma(t)}(A)$$

  (понятен ли смысл этого?).
- Первые шаги очевидны:

$$v[v \mapsto t] = t$$
$$u[v \mapsto t] = u \ (u \text{ — переменная, кроме } v)$$
$$c[v \mapsto t] = c \ (c \text{ — константа})$$
$$f(t_1, \ldots, t_n)[v \mapsto t] = f(t_1[v \mapsto t], \ldots, t_n[v \mapsto t])$$
$$P(t_1, \ldots, t_n)[v \mapsto t] =$$

Подстановки

- Попробуем определить результат замены переменной v на терм t в терм s и в формулу A: $s[v \mapsto t]$ и $A[v \mapsto t]$. Должно получиться

$$\sigma(s[v \mapsto t]) = \sigma_{v \mapsto \sigma(t)}(s)$$
$$\sigma(A[v \mapsto t]) = \sigma_{v \mapsto \sigma(t)}(A)$$

  (понятен ли смысл этого?).
- Первые шаги очевидны:

$$v[v \mapsto t] = t$$
$$u[v \mapsto t] = u \ (u\text{ — переменная, кроме }v)$$
$$c[v \mapsto t] = c \ (c\text{ — константа})$$
$$f(t_1, \ldots, t_n)[v \mapsto t] = f(t_1[v \mapsto t], \ldots, t_n[v \mapsto t])$$
$$P(t_1, \ldots, t_n)[v \mapsto t] = P(t_1[v \mapsto t], \ldots, t_n[v \mapsto t])$$

Подстановки

- Попробуем определить результат замены переменной v на терм t в терм s и в формулу A: $s[v \mapsto t]$ и $A[v \mapsto t]$. Должно получиться

$$\sigma(s[v \mapsto t]) = \sigma_{v \mapsto \sigma(t)}(s)$$
$$\sigma(A[v \mapsto t]) = \sigma_{v \mapsto \sigma(t)}(A)$$

  (понятен ли смысл этого?).

- Первые шаги очевидны:

$$v[v \mapsto t] = t$$
$$u[v \mapsto t] = u \text{ } (u \text{ — переменная, кроме } v)$$
$$c[v \mapsto t] = c \text{ } (c \text{ — константа})$$
$$f(t_1, \ldots, t_n)[v \mapsto t] = f(t_1[v \mapsto t], \ldots, t_n[v \mapsto t])$$
$$P(t_1, \ldots, t_n)[v \mapsto t] = P(t_1[v \mapsto t], \ldots, t_n[v \mapsto t])$$
$$(\neg A)[v \mapsto t] = \neg(A[v \mapsto t]) \text{ (аналогично для } \wedge / \vee / \to)$$

Подстановки с кванторами

- Для формул с кванторами всё сложнее.
- Если квантор по той же переменной:

$$(\forall v\ A)[v \mapsto t] =$$

Подстановки с кванторами

- Для формул с кванторами всё сложнее.
- Если квантор по той же переменной:

$$(\forall v\ A)[v \mapsto t] = \forall v\ A$$

Формула не меняется!

Подстановки с кванторами

- Для формул с кванторами всё сложнее.
- Если квантор по той же переменной:

$$(\forall v\ A)[v \mapsto t] = \forall v\ A$$

  Формула не меняется!
- Если переменные разные, можно предположить

$$(\forall u\ A)[v \mapsto t] =$$

Подстановки с кванторами

- Для формул с кванторами всё сложнее.
- Если квантор по той же переменной:

$$(\forall v\ A)[v \mapsto t] = \forall v\ A$$

  Формула не меняется!
- Если переменные разные, можно предположить

$$(\forall u\ A)[v \mapsto t] = \forall u\ (A[v \mapsto t])$$

Подстановки с кванторами

- Для формул с кванторами всё сложнее.
- Если квантор по той же переменной:

$$(\forall v\ A)[v \mapsto t] = \forall v\ A$$

  Формула не меняется!
- Если переменные разные, можно предположить

$$(\forall u\ A)[v \mapsto t] = \forall u\ (A[v \mapsto t])$$

- Но рассмотрим пример «x делится на 2» $\equiv \exists y\ x = 2 \cdot y$.
  При замене x на y должно получиться

Подстановки с кванторами

- Для формул с кванторами всё сложнее.
- Если квантор по той же переменной:

$$(\forall v\ A)[v \mapsto t] = \forall v\ A$$

  Формула не меняется!
- Если переменные разные, можно предположить

$$(\forall u\ A)[v \mapsto t] = \forall u\ (A[v \mapsto t])$$

- Но рассмотрим пример «x делится на 2» $\equiv \exists y\ x = 2 \cdot y$. При замене x на y должно получиться «y делится на 2». Если попробуем правило выше, получится

Подстановки с кванторами

- Для формул с кванторами всё сложнее.
- Если квантор по той же переменной:

$$(\forall v\ A)[v \mapsto t] = \forall v\ A$$

  Формула не меняется!
- Если переменные разные, можно предположить

$$(\forall u\ A)[v \mapsto t] = \forall u\ (A[v \mapsto t])$$

- Но рассмотрим пример «x делится на 2» $\equiv \exists y\ x = 2 \cdot y$. При замене x на y должно получиться «y делится на 2». Если попробуем правило выше, получится $\exists y\ y = 2 \cdot y$,

Подстановки с кванторами

- Для формул с кванторами всё сложнее.
- Если квантор по той же переменной:

$$(\forall v\ A)[v \mapsto t] = \forall v\ A$$

  Формула не меняется!
- Если переменные разные, можно предположить

$$(\forall u\ A)[v \mapsto t] = \forall u\ (A[v \mapsto t])$$

- Но рассмотрим пример «x делится на 2» $\equiv \exists y\ x = 2 \cdot y$. При замене x на y должно получиться «y делится на 2». Если попробуем правило выше, получится $\exists y\ y = 2 \cdot y$, явно не выражающая «y делится на 2».
- Если в A связаны какие-то переменные терма t, то перед подстановкой их нужно сначала переименовать:

$$(\exists y\ x = 2 \cdot y)[x \mapsto y] = (\exists y'\ x = 2 \cdot y')[x \mapsto y] = \exists y'\ y = 2 \cdot y'$$