

Математическая логика и теория алгоритмов

ЛЕКТОР:
Владимир Анатольевич Захаров

zakh@cs.msu.su

Лекция 1.

Что изучает логика?

Логика в информатике.

Структура курса.

Исторические сведения.

Логические парадоксы.

Что изучает логика?

ЛОГИКА — междисциплинарная отрасль наук,
изучающая

- ▶ законы причинно-следственной связи в окружающем мире;

Что изучает логика?

ЛОГИКА — междисциплинарная отрасль наук, изучающая

- ▶ законы причинно-следственной связи в окружающем мире;
- ▶ проявление законов причинно-следственной связи в рациональном мышлении человека (законы правильного мышления);

Что изучает логика?

ЛОГИКА — междисциплинарная отрасль наук, изучающая

- ▶ законы причинно-следственной связи в окружающем мире;
- ▶ проявление законов причинно-следственной связи в рациональном мышлении человека (законы правильного мышления);
- ▶ отражение законов причинно-следственной связи в языках (естественных и искусственных).

Что изучает логика?

ФОРМАЛЬНАЯ ЛОГИКА

изучает **формы**, в которых проявляются законы причинно-следственных связей, вне зависимости от содержания (смысла) тех явлений (предметов), к которым эти законы относятся.

Что изучает логика?

Поясняющий пример.

P1: Каждый металл — проводник.

P2: Ртуть — металл.

Значит, ртуть — проводник.

Что изучает логика?

Поясняющий пример.

P1: Каждый металл — проводник.

P2: Ртуть — металл.

Значит, ртуть — проводник.

Закон физики (?)

Что изучает логика?

Поясняющий пример.

P1: В каждом южном городе летом тепло.

P2: Севастополь — южный город.

Значит, в Севастополе летом тепло.

Что изучает логика?

Поясняющий пример.

P1: В каждом южном городе летом тепло.

P2: Севастополь — южный город.

Значит, в Севастополе летом тепло.

Закон географии (?)

Что изучает логика?

Поясняющий пример.

P1: Каждый преступник должен быть наказан.

P2: Майор милиции Е. — преступник.

Значит, майор милиции Е. должен быть наказан.

Что изучает логика?

Поясняющий пример.

P1: Каждый преступник должен быть наказан.

P2: Майор милиции Е. — преступник.

Значит, майор милиции Е. должен быть наказан.

Закон юриспруденции (?)

Что изучает логика?

Поясняющий пример.

Общая форма всех этих законов

P1: Каждый предмет, обладающий
свойством **R**, обладает свойством **Q**.

P2: Предмет **c** обладает свойством **R**.

Значит, предмет **c** обладает свойством **Q**.

Что изучает логика?

Поясняющий пример.

Общая форма всех этих законов

P1: Каждый предмет, обладающий свойством **R**, обладает свойством **Q**.

P2: Предмет **c** обладает свойством **R**.

Значит, предмет **c** обладает свойством **Q**.

Закон формальной логики (!!!)

Что изучает логика?

Поясняющий пример.

Общая форма всех этих законов

$$P1: \forall x (R(x) \rightarrow Q(x)).$$

$$P2: R(c).$$

$$Q(c).$$

Закон формальной логики
(в символьном виде)

Логика в информатике

Обратимся вновь к примеру.

Это — исходные знания (база знаний).

P1: Каждый металл — проводник.

P2: Ртуть — металл.

Логика в информатике

Обратимся вновь к примеру.

Это — исходные знания (база знаний).

P1: Каждый металл — проводник.

P2: Ртуть — металл.

А это — новые знания.

Ртуть — проводник.

Логика в информатике

Обратимся вновь к примеру.

Это — исходные знания (база знаний).

P1: Каждый металл — проводник.

P2: Ртуть — металл.

А это — новые знания.

Ртуть — проводник.

Откуда взялись новые знания???

Логика в информатике

Применение закона формальной логики.

Логический закон:

$P1: \forall x (R(x) \rightarrow Q(x)).$

$P2: R(c).$

$Q(c).$

Интерпретация:

$R(x)$ — «предмет x — металл»;

$Q(x)$ — «предмет x — проводник»;

c — «ртуть».

Логика в информатике

Еще одно применение закона формальной логики.

Логический закон:

$P1: \forall x (R(x) \rightarrow Q(x)).$

$P2: R(c).$

$Q(c).$

Другая интерпретация:

$R(x)$ — «предмет x — южный город»;

$Q(x)$ — «предмет x — теплый летом»;

c — «Севастополь».

Логика в информатике

Логика не позволяет получать новую
информацию!!!

Логика в информатике

Логика не позволяет получать новую информацию!!!

Знания — это представление информации в виде формальных высказываний.

Законы формальной логики преобразуют одни высказывания в другие.

Таким образом, законы формальной логики позволяют преобразовывать информацию из одной формы представления в другую.

Законы формальной логики — это инструмент преобразования информации.

Логика в информатике

Основная задача формальной логики.

База знаний: $\Gamma = \{\varphi_1, \varphi_2, \dots, \varphi_N\}$.

Предложение: ψ .

Задача (неформальная): выяснить, является ли предложение ψ следствием утверждений базы знаний Γ .

Задача (формальная): проверить, что ψ выводится из Γ по законам формальной логики.

Логика в информатике

Приложение 1.

Экспертные системы.

База знаний Γ — база знаний экспертной системы.

Предложение ψ — запрос к базе знаний.

Аппарат логического вывода — ядро экспертной системы.

Логика в информатике

Приложение 1.

Экспертные системы.

База знаний Γ — база знаний экспертной системы.

Предложение ψ — запрос к базе знаний.

Аппарат логического вывода — ядро экспертной системы.

Приложение 2.

Автоматизация научных исследований.

База знаний Γ — система аксиом математической теории.

Предложение ψ — математическое утверждение.

Аппарат логического вывода — ядро автоматической системы доказательства теорем.

Логика в информатике

Приложения 1, 2.

Для этого нужно уметь:

- ▶ Создать формальный язык для представления знаний.

Логика в информатике

Приложения 1, 2.

Для этого нужно уметь:

- ▶ Создать формальный язык для представления знаний.
- ▶ Выделить необходимую систему законов формальной логики.

Логика в информатике

Приложения 1, 2.

Для этого нужно уметь:

- ▶ Создать формальный язык для представления знаний.
- ▶ Выделить необходимую систему законов формальной логики.
- ▶ Проверить корректность логических законов.

Логика в информатике

Приложения 1, 2.

Для этого нужно уметь:

- ▶ Создать формальный язык для представления знаний.
- ▶ Выделить необходимую систему законов формальной логики.
- ▶ Проверить корректность логических законов.
- ▶ Проверить полноту построенной системы логических законов.

Логика в информатике

Приложения 1, 2.

Для этого нужно уметь:

- ▶ Создать формальный язык для представления знаний.
- ▶ Выделить необходимую систему законов формальной логики.
- ▶ Проверить корректность логических законов.
- ▶ Проверить полноту построенной системы логических законов.
- ▶ Разработать алгоритм проверки выводимости одних предложений из других по заданным логическим законам.

Логика в информатике

Приложения 1, 2.

Для этого нужно уметь:

- ▶ Создать формальный язык для представления знаний.
- ▶ Выделить необходимую систему законов формальной логики.
- ▶ Проверить корректность логических законов.
- ▶ Проверить полноту построенной системы логических законов.
- ▶ Разработать алгоритм проверки выводимости одних предложений из других по заданным логическим законам.

Этим задачам посвящена первая часть курса.

Логика в информатике

Приложения 1, 2.

Для этого нужно уметь:

- ▶ Создать формальный язык для представления знаний.
- ▶ Выделить необходимую систему законов формальной логики.
- ▶ Проверить корректность логических законов.
- ▶ Проверить полноту построенной системы логических законов.
- ▶ Разработать алгоритм проверки выводимости одних предложений из других по заданным логическим законам.

Этим задачам посвящена первая часть курса.

- ▶ Оптимизировать построенный алгоритм (сделать его практически пригодным).

Этой задаче посвящена вторая часть курса.

Логика в информатике

Приложение 3.

Программирование.

Вычисление программы — последовательное преобразование интерпретатором одних состояний данных в другие согласно заданному алгоритму .

Логический вывод (доказательство) — последовательное построение по законам формальной логики одних утверждений из других, исходя из заданной базы знаний .

Логика в информатике

Приложение 3.

Программирование.

Вычисление программы — последовательное преобразование интерпретатором одних состояний данных в другие согласно заданному алгоритму .

Логический вывод (доказательство) — последовательное построение по законам формальной логики одних утверждений из других, исходя из заданной базы знаний .

Программа

База знаний Г

Логика в информатике

Приложение 3.

Программирование.

Вычисление программы — последовательное преобразование интерпретатором одних состояний данных в другие согласно заданному алгоритму .

Логический вывод (доказательство) — последовательное построение по законам формальной логики одних утверждений из других, исходя из заданной базы знаний .

Программа

База знаний Γ

Вызов программы

Запрос ψ

Логика в информатике

Приложение 3.

Программирование.

Вычисление программы — последовательное преобразование интерпретатором одних состояний данных в другие согласно заданному алгоритму .

Логический вывод (доказательство) — последовательное построение по законам формальной логики одних утверждений из других, исходя из заданной базы знаний .

Программа

Вызов программы

Вычисление программы

База знаний Γ

Запрос ψ

Логический вывод
(доказательство)

Логика в информатике

Приложение 3.

Программирование.

Вычисление программы — последовательное преобразование интерпретатором одних состояний данных в другие согласно заданному алгоритму .

Логический вывод (доказательство) — последовательное построение по законам формальной логики одних утверждений из других, исходя из заданной базы знаний .

Программа

Вызов программы

Вычисление программы

Интерпретатор программ

База знаний Γ

Запрос ψ

Логический вывод
(доказательство)

Правила вывода

Логика в информатике

Приложение 3.

Программирование.

Однако не все так просто.

Ведь успешное вычисление программы завершается результатом, в то время как не всякое доказательство, даже если оно корректно, является «результативным» (конструктивным).

Многие теоремы существования можно успешно доказать, не предъявив при этом искомого объекта.

Логика в информатике

Приложение 3.

Пример.

Задача. Существуют ли такие иррациональные числа α и β , что α^β — рациональное число?

Логика в информатике

Приложение 3.

Пример.

Задача. Существуют ли такие иррациональные числа α и β , что α^β — рациональное число?

Решение.

1. Если $\sqrt{2}^{\sqrt{2}}$ — рациональное число, то $\alpha = \beta = \sqrt{2}$.

Логика в информатике

Приложение 3.

Пример.

Задача. Существуют ли такие иррациональные числа α и β , что α^β — рациональное число?

Решение.

1. Если $\sqrt{2}^{\sqrt{2}}$ — рациональное число, то $\alpha = \beta = \sqrt{2}$.
2. Если $\sqrt{2}^{\sqrt{2}}$ — иррациональное число, то $\alpha = \sqrt{2}^{\sqrt{2}}$, $\beta = \sqrt{2}$.

Логика в информатике

Приложение 3.

Пример.

Задача. Существуют ли такие иррациональные числа α и β , что α^β — рациональное число?

Решение.

1. Если $\sqrt{2}^{\sqrt{2}}$ — рациональное число, то $\alpha = \beta = \sqrt{2}$.
2. Если $\sqrt{2}^{\sqrt{2}}$ — иррациональное число, то $\alpha = \sqrt{2}^{\sqrt{2}}$, $\beta = \sqrt{2}$.

Таким образом, искомые числа α и β существуют. \square

Логика в информатике

Приложение 3.

Пример.

Задача. Существуют ли такие иррациональные числа α и β , что α^β — рациональное число?

Решение.

1. Если $\sqrt{2}^{\sqrt{2}}$ — рациональное число, то $\alpha = \beta = \sqrt{2}$.
2. Если $\sqrt{2}^{\sqrt{2}}$ — иррациональное число, то $\alpha = \sqrt{2}^{\sqrt{2}}$, $\beta = \sqrt{2}$.

Таким образом, искомые числа α и β существуют. \square

Мы доказали существование нужных чисел α и β , но не смогли их **вычислить**. Это **неконструктивное** доказательство.

Чтобы логическое доказательство могло играть роль вычисления, оно должно быть конструктивным.

Логика в информатике

Приложение 3.

Для этого нужно уметь:

- ▶ Разработать формальный язык для представления программ в виде логических утверждений.

Логика в информатике

Приложение 3.

Для этого нужно уметь:

- ▶ Разработать формальный язык для представления программ в виде логических утверждений.
- ▶ Сделать логическое доказательство конструктивным, чтобы оно могло играть роль вычисления.

Логика в информатике

Приложение 3.

Для этого нужно уметь:

- ▶ Разработать формальный язык для представления программ в виде логических утверждений.
- ▶ Сделать логическое доказательство конструктивным, чтобы оно могло играть роль вычисления.
- ▶ Проверить вычислительную корректность этого способа доказательства.

Логика в информатике

Приложение 3.

Для этого нужно уметь:

- ▶ Разработать формальный язык для представления программ в виде логических утверждений.
- ▶ Сделать логическое доказательство конструктивным, чтобы оно могло играть роль вычисления.
- ▶ Проверить вычислительную корректность этого способа доказательства.
- ▶ Проверить алгоритмическую полноту этого способа доказательства.

Логика в информатике

Приложение 3.

Для этого нужно уметь:

- ▶ Разработать формальный язык для представления программ в виде логических утверждений.
- ▶ Сделать логическое доказательство конструктивным, чтобы оно могло играть роль вычисления.
- ▶ Проверить вычислительную корректность этого способа доказательства.
- ▶ Проверить алгоритмическую полноту этого способа доказательства.
- ▶ Сделать этот способ программирования удобным для пользования.

Этому посвящена третья часть курса.

Логика в информатике

Приложение 4.

Программы могут быть правильными и неправильными.

Правильная программа — это такая программа, поведение которой удовлетворяет заданным требованиям (спецификации) корректности.

Проверить правильность программы — значит доказать, что программа удовлетворяет требованиям корректности.

Для доказательства правильности программ могут быть использованы методы логики.

Логика в информатике

Приложение 4.

Для этого нужно уметь:

- ▶ Создать формальный язык для описания требований правильности программ.

Логика в информатике

Приложение 4.

Для этого нужно уметь:

- ▶ Создать формальный язык для описания требований правильности программ.
- ▶ Разработать правила логического доказательства правильности программ.

Логика в информатике

Приложение 4.

Для этого нужно уметь:

- ▶ Создать формальный язык для описания требований правильности программ.
- ▶ Разработать правила логического доказательства правильности программ.
- ▶ Разработать алгоритм (метод) применения этих правил для доказательства правильности программ относительно заданных спецификаций.

Логика в информатике

Приложение 4.

Для этого нужно уметь:

- ▶ Создать формальный язык для описания требований правильности программ.
- ▶ Разработать правила логического доказательства правильности программ.
- ▶ Разработать алгоритм (метод) применения этих правил для доказательства правильности программ относительно заданных спецификаций.

Этому посвящена четвертая часть курса.

И кто же это все
придумал?

Исторические сведения



384 д.н.э.

322 д.н.э.

АРИСТОТЕЛЬ

Исторические сведения

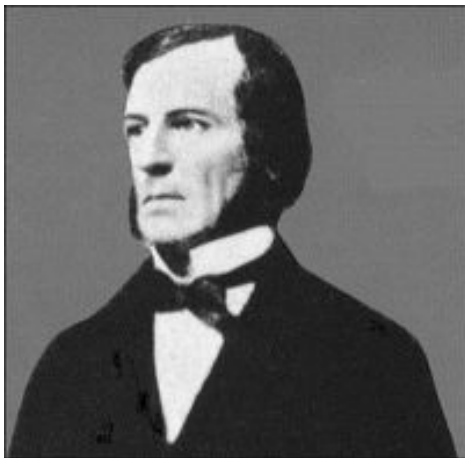


1646

1716

ГОТФРИД ВИЛЬГЕЛЬМ ФОН ЛЕЙБНИЦ

Исторические сведения

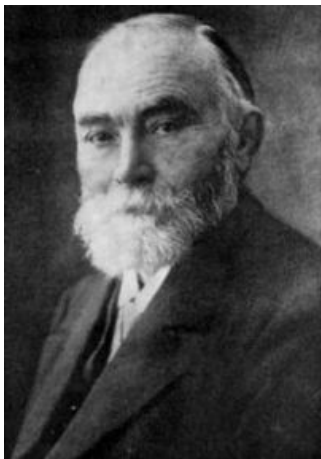


1815

1864

ДЖОРДЖ БУЛЬ

Исторические сведения



1848

1925

ГОТТЛОБ ФРЕГЕ

Исторические сведения

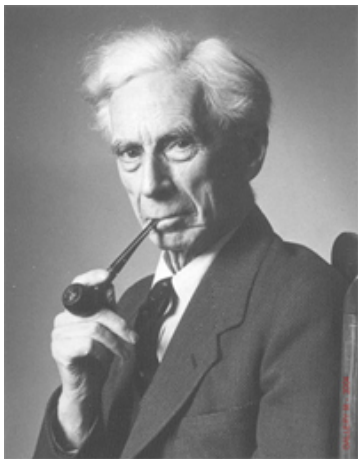


1862

1943

ДАВИД ГИЛЬБЕРТ

Исторические сведения

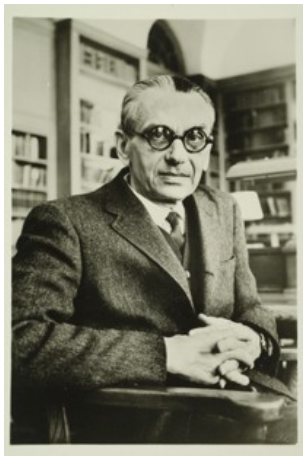


1872

1970

БЕРТРАН РАССЕЛ

Исторические сведения



1906

1978

КУРТ ГЕДЕЛЬ

Исторические сведения



1902

1983

АЛЬФРЕД ТАРСКИЙ

Исторические сведения



1908

1931

ЖАК ЭРБРАН

Исторические сведения

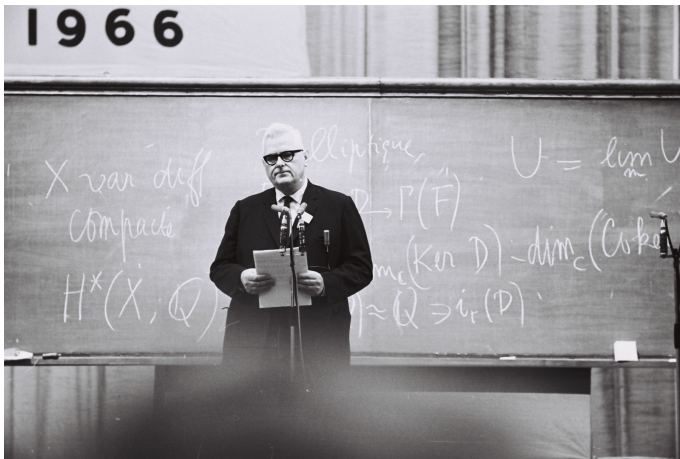


1909

1945

ГЕРХАРД ГЕНЦЕН

Исторические сведения



1903

1995

АЛОНЗО ЧЕРЧ

Исторические сведения



1909

1994

СТИВЕН КЛИНИ

Исторические сведения



1909

1967

АНАТОЛИЙ ИВАНОВИЧ МАЛЬЦЕВ

Исторические сведения

1909

1967

АНАТОЛИЙ ИВАНОВИЧ МАЛЬЦЕВ

Откуда произошла
логика?

Логические парадоксы.

Парадоксы - движущая сила логики.

Противоречивые парадоксы (**антиномии**) заставляют задумываться над такими вопросами, как

- ▶ Что такое истинное утверждение?
- ▶ Что такое доказуемое утверждение?
- ▶ В какой мере можно «механизировать» рассудочное мышление?

Логические парадоксы.

Парадокс о крокодиле

Крокодил схватил ребенка.

Мать ребенка просит крокодила:

«Верни мне ребенка!»

Крокодил отвечает:

«Я верну тебе ребенка, если ты угадаешь, исполню ли я твою просьбу.»

«Не исполнишь,» — говорит женщина.

Логические парадоксы.

Парадокс о крокодиле

Крокодил схватил ребенка.

Мать ребенка просит крокодила:

«Верни мне ребенка!»

Крокодил отвечает:

«Я верну тебе ребенка, если ты угадаешь, исполню ли я твою просьбу.»

«Не исполнишь,» — говорит женщина.

Вернет ли крокодил ребенка матери?

Логические парадоксы.

Парадокс лжеца (6 век д.н.э.)

УТВЕРЖДЕНИЕ,
ИЗОБРАЖЕННОЕ НА ЭТОМ
СЛАЙДЕ, — ЛОЖНОЕ.

Логические парадоксы.

Парадокс лжеца (6 век д.н.э.)

Истинно или ложно предъявленное вам утверждение?

Логические парадоксы.

Парадокс утренней звезды

Венера видна **ранним вечером**, и поэтому ее называют **вечерней звездой**.

Венера видна **ранним утром**, и поэтому ее называют **утренней звездой**.

Логические парадоксы.

Парадокс утренней звезды

Венера видна **ранним вечером**, и поэтому ее называют **вечерней звездой**.

Венера видна **ранним утром**, и поэтому ее называют **утренней звездой**.

Означает ли это, что **вечерняя звезда** видна **ранним утром** ?

Логические парадоксы.

Парадокс морской битвы

Некий флотоводец обратился к прорицателю с вопросом, состоится ли завтра морская битва. Прорицатель ответил: «Битва завтра состоится».

На следующий день случился шторм, и флот не смог выйти в море. Разгневанный флотоводец потребовал от прорицателя вернуть деньги, поскольку его прогноз оказался ложным. Прорицатель ответил: «Твои моряки вчера купили на рынке свежее молоко. Сегодня это молоко уже не свежее, но они не просят вернуть им деньги обратно. Мой прогноз тоже был верным вчера, и ты не вправе жаловаться на то, что он неверен сегодня».

Логические парадоксы.

Парадокс морской битвы

Некий флотоводец обратился к прорицателю с вопросом, состоится ли завтра морская битва. Прорицатель ответил: «Битва завтра состоится».

На следующий день случился шторм, и флот не смог выйти в море. Разгневанный флотоводец потребовал от прорицателя вернуть деньги, поскольку его прогноз оказался ложным. Прорицатель ответил: «Твои моряки вчера купили на рынке свежее молоко. Сегодня это молоко уже не свежее, но они не просят вернуть им деньги обратно. Мой прогноз тоже был верным вчера, и ты не вправе жаловаться на то, что он неверен сегодня».

Прав ли прорицатель?

Парадоксы неизбежны.

Парадоксы неизбежны.
Но их влияние можно ограничить.

Парадоксы неизбежны.
Но их влияние можно ограничить.
Для этого нужны математические
модели логических законов.

Парадоксы неизбежны.
Но их влияние можно ограничить.
Для этого нужны математические
модели логических законов.
Вот так и появилась

Парадоксы неизбежны.
Но их влияние можно ограничить.
Для этого нужны математические
модели логических законов.
Вот так и появилась

Математическая логика

РЕЙМОНД С. СМАЛЛИАН. "Как же называется эта книга?"

У Порции (героини комедии "Венецианский купец") было три шкатулки: из золота, серебра и свинца. В одной из шкатулок хранился портрет Порции. Поклоннику предлагалось выбрать шкатулку, и если он был достаточно умен, чтобы выбрать шкатулку с портретом, то получал право назвать Порцию невестой. На крышках шкатулок были надписи:

На золотой	На серебряной	На свинцовой
Портрет в этой шкатулке	Портрет не в этой шкатулке	Портрет не в золотой шкатулке

Своему поклоннику Порция пояснила, что из трех высказываний на крышках шкатулок, по крайней мере два ложны.

Какую шкатулку следует выбрать поклоннику Порции?

Загадка Альберта Эйнштейна

В 5 разноцветных домах живут граждане 5 разных стран, которые курят 5 разных сортов сигарет, пьют 5 разных напитков и содержат 5 разных животных. Известно, что

1. У поляка красный дом.
2. Немец курит Rothmans.
3. Курильщик Marlboro живет рядом с тем, кто держит кошку.
4. Финн пьет чай.
5. Курильщик Camel держит попугая.
6. Зеленый дом стоит слева от белого.
7. Курильщик Kent пьет пиво.
8. Грек живет в первом доме.
9. Владелец лошади живет рядом с тем, кто курит Dunhill.
10. Хозяин среднего дома пьет молоко.
11. Хозяин желтого дома курит Dunhill.
12. Хозяин зеленого дома пьет кофе.
13. Швед держит собаку.
14. Грек живет около голубого дома.
15. Курильщик Marlboro живет рядом с тем, кто пьет воду.

Загадка Альберта Эйнштейна

В 5 разноцветных домах живут граждане 5 разных стран, которые курят 5 разных сортов сигарет, пьют 5 разных напитков и содержат 5 разных животных. Известно, что

1. У поляка красный дом.
2. Немец курит Rothmans.
3. Курильщик Marlboro живет рядом с тем, кто держит кошку.
4. Финн пьет чай.
5. Курильщик Camel держит попугая.
6. Зеленый дом стоит слева от белого.
7. Курильщик Kent пьет пиво.
8. Грек живет в первом доме.
9. Владелец лошади живет рядом с тем, кто курит Dunhill.
10. Хозяин среднего дома пьет молоко.
11. Хозяин желтого дома курит Dunhill.
12. Хозяин зеленого дома пьет кофе.
13. Швед держит собаку.
14. Грек живет около голубого дома.
15. Курильщик Marlboro живет рядом с тем, кто пьет воду.

Кому принадлежит золотая рыбка?

КОНЕЦ ЛЕКЦИИ 1.