

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ  
ФЕДЕРАЦИИ

Набережночелнинский институт (филиал)  
ФГАОУ ВПО «Казанский (Приволжский) федеральный университет»

Отделение информационных технологий и энергетических систем

Кафедра системного анализа и информатики

# **Основы интеллектуальных систем для гуманитариев**

## **Лабораторный практикум**

Набережные Челны  
2017

**УДК 004.89(076.5)**

**ББК 32.813я73-5**

**О-75**

Рецензенты:

**Р.Г. Марданшин**, к. ф.-м. н., доцент каф. ЕНД НЧФ КНИТУ-КАИ;

**В.С. Каримов**, к. т. н., доцент каф. системного анализа и информатики НЧИ КФУ.

**Основы интеллектуальных систем для гуманитариев:**

О-75 лабораторный практикум / авт.-сост. : **Л.Ю. Грудцына** ; НЧИ КФУ.  
– Набережные Челны : Издательско-полиграфич. центр, 2017. – 50с. :  
ил., табл. – Библиогр. : 7 назв.

Данный лабораторный практикум предназначен для практического изучения дисциплины «Основы интеллектуальных систем» студентами, обучающимися по направлению подготовки 37.03.01 «Психология».

Практикум состоит из пяти лабораторных работ и содержит краткие теоретические сведения по изучаемой дисциплине. Каждая лабораторная работа включает в себя разобранный пример, а также варианты для самостоятельного выполнения.

**УДК 004.89(076.5)**

**ББК 32.813я73-5**

© НЧИ КФУ, 2017.

© Л.Ю. Грудцына, 2017.

## ОГЛАВЛЕНИЕ

<b>ВВЕДЕНИЕ</b> .....	<b>4</b>
<i>Базовые понятия и основные направления исследований в области искусственного интеллекта</i> .....	4
<i>Представление знаний в системах искусственного интеллекта</i> .....	7
<b>Лабораторная работа №1 «Представление знаний. Продукционная модель»</b> .....	<b>9</b>
<i>Пример решения задачи</i> .....	9
<i>Варианты заданий</i> .....	12
<b>Лабораторная работа 2 «Представление знаний. Семантическая сеть»</b> ..	<b>14</b>
<i>Пример решения задачи</i> .....	15
<i>Варианты заданий</i> .....	18
<b>Лабораторная работа 3 «Представление знаний. Фреймовая модель»</b> .....	<b>19</b>
<i>Пример решения задачи</i> .....	22
<i>Варианты заданий</i> .....	30
<b>Лабораторная работа 4 «Элементы логики высказываний. Способы доказательства и вывода»</b> .....	<b>31</b>
<i>Пример решения задачи</i> .....	34
<i>Варианты заданий</i> .....	37
<b>Лабораторная работа 5 «Элементы нечеткой логики. Нечеткий вывод»</b> ..	<b>39</b>
<i>Пример решения задачи</i> .....	42
<i>Варианты заданий</i> .....	46
<b>ЛИТЕРАТУРА</b> .....	<b>50</b>

## ВВЕДЕНИЕ

### *Базовые понятия и основные направления исследований в области искусственного интеллекта*

**Интеллект** – внутреннее свойство человеческой личности (или искусственной системы), позволяющее ей принимать правильные решения в условиях неопределенности внешней среды.

**Искусственный интеллект** (англ. – *artificial intelligence*) – это искусственные программные системы, созданные человеком на базе ЭВМ и имитирующие решение человеком сложных творческих задач в процессе его жизнедеятельности.

Термин искусственный интеллект (ИИ) был предложен в 1956 г. на семинаре в Дартмутском колледже. На сегодняшний день существует множество определений, описывающих данное понятие.

Чтобы пояснить понятие «искусственный интеллект», необходимо понимать отличие интеллектуальной задачи от простой. Принято считать, если для задачи найден алгоритм её решения, то она не относится к интеллектуальным. Это связано с тем, что, имея алгоритм, процесс решения данного класса задач становится таким, что его может в точности выполнить человек или компьютер, не имеющие ни малейшего представления о сущности самой задачи. С другой стороны, отыскание алгоритма для задач некоторого типа связано со сложными рассуждениями, требующими большой изобретательности и высокой квалификации. Таким образом, **интеллектуальные задачи** относятся к классу трудноформализуемых задач, т.к.:

- в этих задачах используется, помимо традиционных данных в числовом формате, информация в виде изображений, рисунков, знаков, букв, слов, звуков;
- предполагается сделать выбор между многими вариантами в условиях неопределённости;
- не существует алгоритмического решения задачи;
- алгоритмическое решение существует, но его нельзя использовать из-за ограниченности ресурсов.

Искусственный интеллект, как научное направление, в рамках которого ставятся и решаются задачи моделирования интеллектуальной человеческой деятельности, опирается на ряд дисциплин (рис. 1).



Рис. 1. Дисциплины, являющиеся основой ИИ.

С конца 1970-х годов для поддержки решения интеллектуальных задач, традиционно выполнявшихся людьми, на основе исследований в области ИИ сформировалась новая отрасль компьютерной индустрии – разработка интеллектуальных информационных систем.

**Информационная система** – взаимосвязанная совокупность средств, методов и персонала, используемых для хранения, обработки и выдачи информации в интересах достижения поставленной цели.

**Интеллектуальная информационная система (ИИС)** должна уметь в наборе фактов распознать существенные и из имеющихся фактов и знаний сделать выводы с использованием дедукции, аналогии, индукции и т.д. Кроме того, она должна обладать средствами оценки результатов собственной работы. С помощью подсистем объяснения она может ответить на вопрос, почему получен тот или иной результат. Наконец, ИИС должна уметь обобщать, улавливая сходство между имеющимися фактами, и накапливать опыт.

Примерами интеллектуальных информационных систем могут быть:

**Экспертные системы** – системы, имитирующие образ действия высококвалифицированных специалистов в конкретных предметных областях;

**Системы общения с ЭВМ на естественном языке** – специализированные диалоговые системы, обеспечивающие «дружественный» интерактивный диалог человека-оператора с ЭВМ;

**Системы обработки сигналов и изображений** – информационные системы, осуществляющие сбор и обработку информации о состоянии объектов, ее анализ (распознавание, интерпретация), прогнозирование и т.д.;

**Системы управления** – управляющие информационные системы, обеспечивающие оценку состояния управляемого объекта (процесса, системы) и принятие решений с целью достижения высокого качества или выбора стратегии функционирования объекта в условиях неопределенности.

История попыток создания искусственного разума насчитывает более 700 лет. Первую зафиксированную в истории попытку создания машины, моделирующей человеческий разум, связывают с именем испанского изобретателя Раймунда Луллия. Развивая традиции учёных своего времени, Луллий сконструировал машину, состоящую из системы кругов, вращая которые можно было получить «формулу истины». По существу она представляла собой механическую экспертную систему. В XVIII в. Лейбниц и Декарт независимо друг от друга продолжили идеи, заложенные Луллием, предложив универсальные языки классификации всех наук. Эти работы можно считать первыми теоретическими работами в области ИИ.

В 40-х гг. XX в. с появлением компьютера ИИ обрёл второе рождение. Произошло выделение ИИ в самостоятельное научное направление. Исследования в области ИИ проводились в трёх направлениях. В рамках первого подхода объектом исследований являются структура и механизмы работы мозга человека, а конечная цель заключается в раскрытии тайн мышления. Второй подход ориентирован на моделирование интеллектуальной деятельности с помощью компьютера. Третий подход основан на симбиозе возможностей естественного и искусственного интеллекта для создания интерактивных ИИС.

На сегодняшний день методы ИИ позволили создать эффективные компьютерные программы в самых разнообразных, ранее считавшихся недоступными для формализации и алгоритмизации, сферах человеческой деятельности, таких как медицина, биология, зоология, социология, культурология, политология, экономика, бизнес, криминалистика и т.п.

Среди важнейших классов задач, которые ставились перед ИИС, следует выделить: доказательство теорем, управление роботами, распознавание образов, машинный перевод и понимание текстов на естественном языке и игровые программы.

### *Представление знаний в системах искусственного интеллекта*

**Данными** называют информацию фактического характера, описывающую объекты, процессы и явления предметной области, а также их свойства.

**Знания** – это закономерности предметной области (принципы, связи, законы), полученные в результате практической деятельности и профессионального опыта, позволяющие специалистам ставить и решать задачи в этой области. Знания описывают не только отдельные факты, но и взаимосвязи между ними, поэтому знания иногда называют структурированными данными.

**База знаний** – совокупность программных средств, обеспечивающих поиск, хранение, преобразование и запись в памяти ЭВМ сложно структурированных информационных единиц – знаний.

Проблема представления знаний в ИИС чрезвычайно актуальна, поскольку их функционирование опирается на знания о проблемной области, хранящиеся на компьютере. В рамках этой проблемы решаются задачи, связанные с формализацией и представлением знаний в ИИС. Для этого разрабатываются специальные модели представления знаний и языки для описания знаний, выделяются различные типы знаний.

Выделяют два вида моделей предоставления знаний: декларативные и процедурные. В **декларативных** моделях предметная область представляется в виде синтаксического описания её состояния. Вывод решений основывается на процедурах поиска в пространстве состояний. В **процедурном** представлении знания содержатся в небольших программах (процедурах), которые определяют поведение ИИС. При этом можно не описывать все возможные состояния среды или объекта для реализации вывода. Достаточно хранить некоторые начальные состояния и процедуры, генерирующие необходимые описания ситуаций и действий.

К типовым декларативным моделям относят семантические сети и фреймы, а типовым процедурным моделям – исчисления высказываний / предикатов, системы продукций, нечёткая логика. На

практике редко удаётся обойтись рамками одной модели при разработке ИИС, поэтому представление знаний получается сложным.

*Семантическая сеть* представляет собой ориентированный граф, вершинами которого являются информационные единицы, имеющие индивидуальные имена. В качестве информационной единицы могут выступать события, действия, обобщённые понятия или свойства объектов. Вершины графа соединяются дугой, если соответствующие информационные единицы находятся в каком-либо отношении.

*Фрейм* представляет собой структуру данных, дающую целостное представление об объектах, явлениях и их типах в виде абстрактных образов. Структура фрейма записывается в виде списка свойств (слов). Каждый фрейм имеет специальный слот, заполненный наименованием представляемой сущности, а другие заполнены значениями разнообразных атрибутов, ассоциирующихся с объектом.

*Логика высказываний* представляет собой формальную систему, элементами которой являются простые высказывания, из простых высказываний образуются сложные с помощью логических знаков (связок). В логике изучается строение сложных высказываний, выраженных формулами, вне зависимости от содержания составляющих их простых высказываний.

*Логика предикатов* является расширением логики высказываний. Основным объектом здесь является переменное высказывание (предикат), истинность и ложность которого зависят от значения его переменных. Язык логики предикатов является более мощным по сравнению с языком логики высказываний. Он пригоден для формализации понятий многих проблемных областей.

*Продукционная модель*, или модель, основанная на правилах, позволяет представить знания в виде предложений типа ЕСЛИ (условие), ТО (действие).

Количественные данные (знания) могут быть неточными. Для учёта неточности лингвистических знаний используется формальный аппарат *нечёткой алгебры*. Одно из главных понятий в нечёткой логике – это понятие лингвистической переменной, которое определяется через нечёткие множества. Нечёткие множества позволяют учитывать субъективные мнения отдельных экспертов.



## Лабораторная работа №1 «Представление знаний. Продукционная модель»

Продукция – это предложение-образец вида «Если, то», по которому осуществляется поиск в базе знаний.

В продукции выделяют левую часть (начинается с «если» и заканчивается перед «то») и правую (начинается после «то»). Левая часть продукции – антецедент – условие выполнения правой часть продукции. Правая часть – консеквент – действие, выполняемое в случае нахождения элементов, удовлетворяющих левой части. Действие может быть промежуточным и выступать затем в качестве консеквента или целевым, завершающим процедуру вывода.

Антецедент формируется из фактов, входных данных задачи и логических связей (и, или, не). Консеквент может представлять из себя действие по изменению фактов, данных, рекомендацию, решение задачи. Кроме этого, любая продукция имеет имя и приоритет, определяющий последовательность проверки продукций машиной вывода.

Продукции отражают причинно-следственные связи, которые и позволяют человеку принимать решения, базируясь на знаниях и предположениях о том, что есть и что будет, если что-то сделать.

### *Пример решения задачи*

**Задача.** Построить продукционную модель представления знаний в предметной области «Ресторан» (посещение ресторана).

**Описание процесса решения.** Для построения продукционной модели представления знаний необходимо выполнить следующие шаги:

- 1) Определить целевые действия задачи (являющиеся решениями).
- 2) Определить промежуточные действия или цепочку действий, между начальным состоянием и конечным (между тем, что имеется, и целевым действием).
- 3) Опередить условия для каждого действия, при котором его целесообразно и возможно выполнить. Определить порядок выполнения действий.
- 4) Добавить конкретики при необходимости, исходя из поставленной задачи.

5) Преобразовать полученный порядок действий и соответствующие им условия в продукции.

6) Для проверки правильности построения продукций записать цепочки продукций, явно проследив связи между ними.

Этот набор шагов предполагает движение при построении производственной модели от результата к начальному состоянию, но возможно и движение от начального состояния к результату (шаги 1 и 2).

### **Решение.**

1) Обязательное действие, выполняемое в ресторанах – поглощение пищи и ее оплата. Значит, есть уже два целевых действия «съесть пищу» и «оплатить», которые взаимосвязаны и следуют друг за другом.

2) Прежде чем что-либо съесть в ресторане, туда нужно придти, дождаться официанта и сделать заказ. Кроме того, нужно выбрать, в какой именно ресторан пойти. Значит, цепочка промежуточных действий: «выбор ресторана и путь туда», «сделать заказ официанту».

3) Прежде чем идти в ресторан, необходимо убедиться, что есть необходимая сумма денег. Выбор ресторана может обуславливаться многими причинами, выберем территориальный признак – к какому ближе в тот и идем. В разных ресторанах работают разные люди, поэтому в зависимости от выбора ресторана, официанты будут разные. Кроме того, разные рестораны специализируются на разных кухнях, поэтому заказанные блюда будут в разных ресторанах отличаться. Значит вначале идут действия, позволяющие выбрать ресторан, затем характеризующие рестораны, а уже после заказ, еда, и оплата заказа.

4) Пусть в задаче будут рассматриваться два ресторана: «Вкусная еда» и «Вкуснятина». Первый – паб и заказы приносят быстрее, чем во втором, второй – пиццерия. В первом работает официант Сергей, а во втором официантка Марина. Петр – это клиент.

5) Выше описанное можно преобразовать в следующие предложения типа «Если, то»:

- *Если субъект хочет есть и у субъекта есть достаточная сумма денег, то субъект может пойти в ресторан.*
- *Если субъект ближе к ресторану «Вкусная еда», чем к ресторану «Вкуснятина» и субъект может пойти в ресторан, то субъект идет в ресторан «Вкусная еда».*

- Если субъект ближе к ресторану «Вкуснятина», чем к ресторану «Вкусная еда» и субъект может пойти в ресторан, то субъект идет в ресторан «Вкуснятина».
- Если субъект идет в ресторан «Вкуснятина» и в ресторане «Вкуснятина» работает официант Марина, то у субъекта принимает заказ Марина.
- Если субъект идет в ресторан «Вкусная еда» и в ресторане «Вкусная еда» работает официант Сергей, то у субъекта принимает заказ Сергей.
- Если субъект выбрал блюда и у субъекта принимает заказ Марина, то заказ принесут через 20 мин.
- Если субъект выбрал блюда и у субъекта принимает заказ Сергей, то заказ принесут через 10 мин.
- Если заказ принесут через 20 мин. или заказ принесут через 10 мин., то субъект может есть.
- Если субъект может есть, то после еды субъект должен оплатить заказ.

Введем обозначения для фактов (Ф), действий (Д) и продукций (П),

тогда:

Субъект = Петр;

Ф1= субъект хочет есть;

Ф2= у субъекта есть достаточная сумма денег;

Ф3= субъект ближе к ресторану «Вкусная еда», чем к «Вкуснятина»;

Ф4=в ресторане «Вкуснятина» работает официант Марина;

Ф5=в ресторане «Вкусная еда» работает официант Сергей;

Ф6= субъект выбрал блюда;

Д1= субъект может пойти в ресторан;

Д2=субъект идет в ресторан «Вкусная еда»;

Д3=субъект идет в ресторан «Вкуснятина»;

Д4= у субъекта принимает заказ Марина;

Д5=у субъекта принимает заказ Сергей;

Д6=заказ принесут через 20 мин.

Д7=заказ принесут через 10 мин.

Д8=после еды субъект должен оплатить заказ.

Для продукций установим приоритет (в скобках перед запятой, чем выше приоритет, чем раньше проверяется правило).

- П1(4 , Ф1 и Ф2)= Д1;
- П2(5 , Ф3 и Д1)= Д2;
- П3(4 , не Ф3 и Д1)= Д3;
- П4(3 , Д3 и Ф4)= Д4;
- П5(3 , Д2 и Ф5)= Д5;
- П6(2 , Д4)= Д6;
- П7(2 , Д5)= Д7;
- П8(1 , Д6 или Д7)= Д8;

6) Для отображения взаимосвязи продукций построим граф (рис. 2).

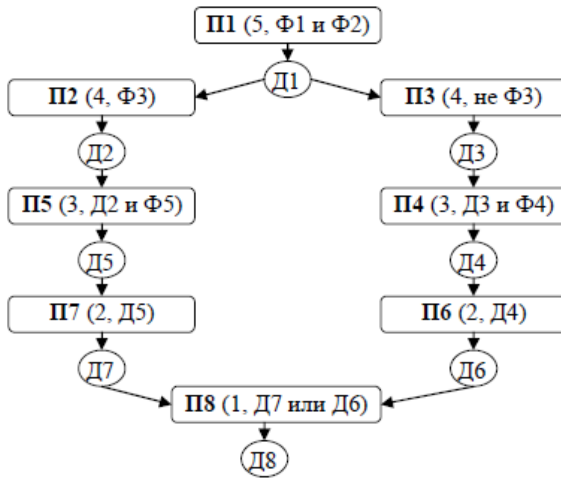


Рис. 2. Схема продукций предметной области «Ресторан».

### Варианты заданий

1. Построить продукционную модель представления знаний в предметной области «Аэропорт» (диспетчерская).
2. Построить продукционную модель представления знаний в предметной области «Железная дорога» (продажа билетов).
3. Построить продукционную модель представления знаний в предметной области «Торговый центр» (организация).
4. Построить продукционную модель представления знаний в предметной области «Автозаправка» (обслуживание клиентов).
5. Построить продукционную модель представления знаний в предметной области «Автопарк» (пассажирыские перевозки).

6. Построить производственную модель представления знаний в предметной области «Компьютерные сети» (организация).
7. Построить производственную модель представления знаний в предметной области «Университет» (учебный процесс).
8. Построить производственную модель представления знаний в предметной области «Компьютерная безопасность» (средства и способы ее обеспечения).
9. Построить производственную модель представления знаний в предметной области «Компьютерная безопасность» (угрозы).
10. Построить производственную модель представления знаний в предметной области «Интернет-кафе» (организация и обслуживание).
11. Построить производственную модель представления знаний в предметной области «Разработка информационных систем» (ведение информационного проекта).
12. Построить производственную модель представления знаний в предметной области «Туристическое агентство» (работа с клиентами).
13. Построить производственную модель представления знаний в предметной области «Зоопарк» (организация).
14. Построить производственную модель представления знаний в предметной области «Кухня» (приготовление пищи).
15. Построить производственную модель представления знаний в предметной области «Больница» (прием больных).
16. Построить производственную модель представления знаний в предметной области «Кинопрокат» (ассортимент и работа с клиентами).
17. Построить производственную модель представления знаний в предметной области «Прокат автомобилей» (ассортимент и работа с клиентами).
18. Построить производственную модель представления знаний в предметной области «Операционные системы» (функционирование).
19. Построить производственную модель представления знаний в предметной области «Информационные системы» (виды и функционирование).
20. Построить производственную модель представления знаний в предметной области «Предприятие» (структура и функционирование).

## Лабораторная работа 2 «Представление знаний. Семантическая сеть»

Семантическая сеть — это ориентированный граф, вершины которого — понятия, а дуги — отношения между ними. Узлы в семантической сети обычно соответствуют объектам, концепциям, событиям или понятиям. Любой фрагмент сети, например одна вершина, две вершины и соединяющие их дуги, называют подсетью. Логический вывод (поиск решения) на семантической сети заключается в том, чтобы найти или сконструировать подсеть, удовлетворяющую некоторым условиям.

Отношения, представляемые дугами, в семантической сети могут быть различными (таблица 2). Типы отношений выбираются в зависимости от вида семантической сети (таблица 3) и решаемой задачи.

Таблица 2. Основные виды отношений в семантических сетях.

Тип	Описание
Являться наследником (a-kind-of)	задает иерархические связи между классами
Являться экземпляром (is-a, например)	определяет значение, описывает конкретный объект, понятие
Это (are, есть)	может использоваться вместо связи a-kind-of в отношениях подразумевающих равенство или эквивалентность
Являться частью (has-part)	определяет структурные связи, описывает части или целые объекты
Функциональные	определяются обычно глаголами, отражают различные отношения (учить, владеть и т.д.)
Количественные	отображают количественные соотношения между вершинами (больше, меньше и т.д.)
Пространственные	отображают пространственные отношения между вершинами (близко, далеко и т.д.)
Временные	описывают временные связи между вершинами (скоро, долго, сейчас и т.д.)
Атрибутивные	описывают свойства объектов, понятий
Логические	описывают логические связи между вершинами (и, или, не)

Таблица 3. Типы семантических сетей.

Тип	Описание
<i>По типу знания</i>	
Экстенциональные	описывает конкретные отношения данной ситуации
Интенциональные	описывают имена классов объектов, а не индивидуальные имена объектов, связи отражают те отношения, которые всегда присущи объектам данного класса
<i>По типу ограничений на дуги и вершины</i>	
Простые	вершины сети не обладают внутренней структурой
Иерархические	вершины обладают внутренней структурой, в иерархической сети есть возможность разделять сеть на подсети и устанавливать отношения не только между вершинами, но и между подсетями (различные подсети, существующие в сети, могут быть упорядочены в виде дерева подсетей, вершины которого—подсети, а дуги — отношения видимости)
Динамические (сценарии)	сети с событиями
<i>По количеству типов отношений</i>	
Однородные	обладают только одним типом отношений
Неоднородные	количество типов отношений больше двух
<i>По арности отношений</i>	
Бинарные	все отношения в графе связывают ровно два понятия
N-арные	в сети есть отношения, связывающие более двух объектов

### *Пример решения задачи*

**Задача.** Построить сетевую модель представления знаний в предметной области «Ресторан» (посещение ресторана).

**Описание процесса решения.** Для построения сетевой модели представления знаний необходимо выполнить следующие шаги:

- 1) Определить абстрактные объекты и понятия предметной области, необходимые для решения поставленной задачи. Оформить их в виде вершин.
- 2) Задать свойства для выделенных вершин, оформив их в виде вершин, связанных с исходными вершинами атрибутивными отношениями.
- 3) Задать связи между этими вершинами, используя функциональные, пространственные, количественные, логические, временные, атрибутивные отношения, а также отношения типа «являться наследником» и «являться частью».
- 4) Добавить конкретные объекты и понятия, описывающие решаемую задачу. Оформить их в виде вершин, связанных с уже существующими отношениями типа «являться экземпляром», «есть».
- 5) Проверить правильность установленных отношений (вершины и само отношение при правильном построении образуют предложение, например «Двигатель является частью автомобиля»).

### **Решение.**

1) Ключевые понятия данной предметной области – ресторан, тот, кто посещает ресторан (клиент) и те, кто его обслуживают (повара, метрдотели, официанты, для простоты ограничимся только официантами). У обслуживающего персонала и клиентов есть общие характеристики, поэтому целесообразно выделить общее абстрактное понятие – человек. Продукцией ресторана являются блюда, которые заказывают клиенты.

Исходя из этого, вершины графа будут следующими: «Ресторан», «Человек», «Официант», «Клиент», «Заказ» и «Блюдо».

2) У этих объектов есть определенные свойства и атрибуты. Например, рестораны располагаются по определенным адресам, каждое блюдо из меню имеет свою цену. Поэтому добавим вершины «Адрес» и «Цена».

3) Определим для имеющихся вершин отношения и их типы, используя

таблицу 2.

4) Добавим знание о конкретных фактах решаемой задачи. Пусть имеется два ресторана: «Вкуснятина» и «Вкусная еда», в первом работает официантка Марина, а во втором официант Сергей. Пётр решил пойти в ресторан «Вкусная еда» и сделал заказ официанту на 2 блюда: картофель фри



за 30 р., бифштекс за 130 р. Также известны адреса этих ресторанов и их специфика.

Исходя из этого, добавим соответствующие вершины в граф и соединим их функциональными отношениями и отношениями типа «например или являться экземпляром». Полученный в результате граф изображен на рис. 3.

5) Осуществим проверку установленных связей. Например, возьмем вершину «Блюдо» и пройдем по установленным связям. Получаем следующую информацию: блюдо является частью заказа, примерами блюд могут служить картофель фри и бифштекс.

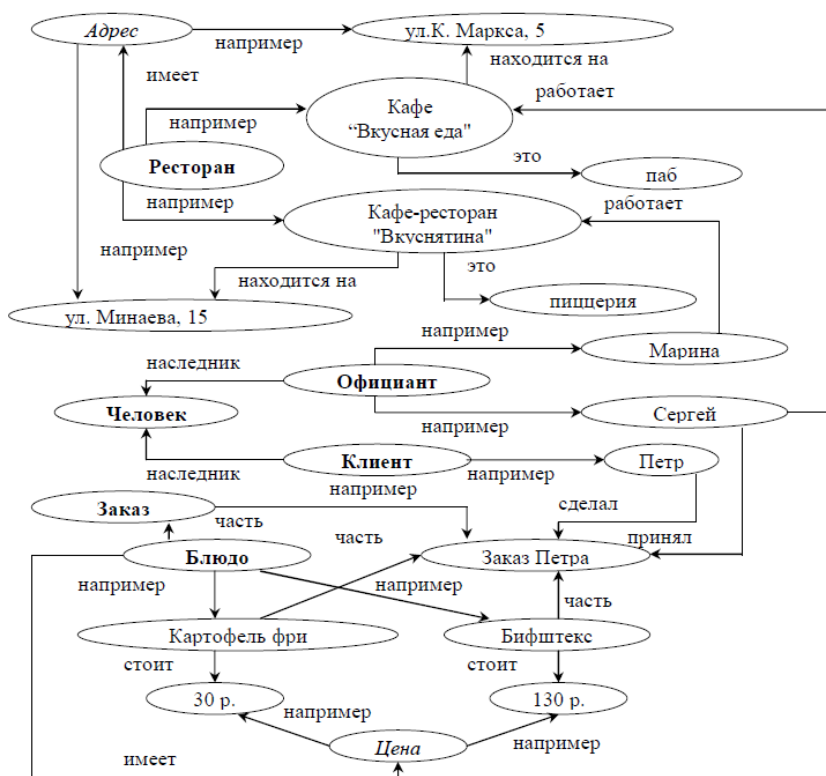


Рис. 3. Семантическая сеть предметной области «Ресторан».

Для получения ответа на какой-либо вопрос по этой задачи, необходимо найти соответствующий участок сети и, используя связи, получить результат.

Например, вопрос «Какова цена заказа Петра (сколько Петр заплатил за заказ)?» Из запроса понятно, что необходимо найти следующие вершины: «Цена», «Петр» и «Заказ» или «Заказ Петра». Часть семантической сети, находящаяся между этими вершинами, содержит ответ, а именно, частью заказа Петра являются картофель фри и бифштекс, которые стоят 30 и 130 р. соответственно. Больше информации о заказе Петра в модели нет, поэтому делаем вывод – Петр заплатил 160 р.

### *Варианты заданий*

1. Построить продукционную модель представления знаний в предметной области «Аэропорт» (диспетчерская).
2. Построить продукционную модель представления знаний в предметной области «Железная дорога» (продажа билетов).
3. Построить продукционную модель представления знаний в предметной области «Торговый центр» (организация).
4. Построить продукционную модель представления знаний в предметной области «Автозаправка» (обслуживание клиентов).
5. Построить продукционную модель представления знаний в предметной области «Автопарк» (пассажирыские перевозки).
6. Построить продукционную модель представления знаний в предметной области «Компьютерные сети» (организация).
7. Построить продукционную модель представления знаний в предметной области «Университет» (учебный процесс).
8. Построить продукционную модель представления знаний в предметной области «Компьютерная безопасность» (средства и способы ее обеспечения).
9. Построить продукционную модель представления знаний в предметной области «Компьютерная безопасность» (угрозы).
10. Построить продукционную модель представления знаний в предметной области «Интернет-кафе» (организация и обслуживание).
11. Построить продукционную модель представления знаний в предметной области «Разработка информационных систем» (ведение информационного проекта).
12. Построить продукционную модель представления знаний в предметной области «Туристическое агентство» (работа с клиентами).

13. Построить производственную модель представления знаний в предметной области «Зоопарк» (организация).
14. Построить производственную модель представления знаний в предметной области «Кухня» (приготовление пищи).
15. Построить производственную модель представления знаний в предметной области «Больница» (прием больных).
16. Построить производственную модель представления знаний в предметной области «Кинопрокат» (ассортимент и работа с клиентами).
17. Построить производственную модель представления знаний в предметной области «Прокат автомобилей» (ассортимент и работа с клиентами).
18. Построить производственную модель представления знаний в предметной области «Операционные системы» (функционирование).
19. Построить производственную модель представления знаний в предметной области «Информационные системы» (виды и функционирование).
20. Построить производственную модель представления знаний в предметной области «Предприятие» (структура и функционирование).

### **Лабораторная работа 3 «Представление знаний. Фреймовая модель»**

Фреймовая модель представления знаний была предложена М. Минским в 1979 году и является развитием семантических сетей.

Фрейм (англ. frame) - абстрактный образ для представления некоторого стереотипа восприятия. Каждый фрейм имеет собственное название и список слотов и их значений.

Значениями могут быть данные любого типа, а также название другого фрейма. Таким образом, фреймы образуют сеть. Кроме того, существует связь между фреймами типа АКО (a kind of), которая указывает на фрейм более высокого уровня иерархии, откуда неявно наследуются список и значения слотов. При этом возможно множественное наследование – перенос свойств от нескольких прототипов.

Любой фрейм может быть представлен следующим образом:

*(ИМЯ ФРЕЙМА:*

*(имя 1-го слота: значение 1-го слота),*

(имя 2-го слота: значение 2-го слота),

.....

(имя N-го слота: значение N-го слота)).

Табличное представление слота выглядит следующим образом (таблица 4):

Таблица 4. Структура фрейма.

ИМЯ ФРЕЙМА			
Имя слота	Значение слота	Способ получения значения	Демон

При табличном представлении фрейма кроме уже описанных составляющих фрейма указываются и дополнительные параметры. Способ получения значения определяет, как именно устанавливается значение конкретного слота. Существует несколько способов (таблица 5), выбор способа зависит от свойств самих данных.

Таблица 5. Способы получения значений слотов.

Способ	Описание
По умолчанию от прототипа (родителя)	Слоту присваивается значение, определенное по умолчанию во фрейме-прототипе, некоторые стандартные значения.
Через наследование	Отличается от первого способа тем, что значение задано в специальном слоте родительского фрейма, соединенного с текущим связью АКО.
По формуле	Слоту назначается формула, результат вычисления которой является значением слота.
Через присоединенную процедуру	Слоту назначается процедура, позволяющая получить значение слота алгоритмически.
Из внешних источников данных	При использовании модели в интеллектуальных системах данные, являющиеся значениями слотов, могут поступать из баз данных, от системы датчиков, от пользователя.

В теории фреймов допускается, чтобы к слотам присоединялись различные специальные процедуры. Для этого используются так называемые демоны. Демоном (таблица 6) называется процедура, автоматически

запускаемая при выполнении некоторого условия (события) при обращении к соответствующему слоту. Демонов может быть несколько. Наиболее похож механизм присоединенных процедур к триггерам в реляционных базах данных.

*Таблица 6. Наиболее распространенные демоны.*

<b>Демон</b>	<b>Событие</b>	<b>Описание</b>
IF-REMOVED	если удалено	Выполняется, когда информация удаляется из слота.
IF-ADDED	если добавлено	Выполняется, когда новая информация записывается в слот.
IF-NEEDED	по требованию	Выполняется, когда запрашивается информация из пустого слота.
IF-DEFAULT	по умолчанию	Выполняется, когда устанавливается значение по умолчанию.

Существует несколько видов фреймов, которые позволяют описать предметную область и решаемую задачу. В таблице 7 представлены наиболее распространенные типы фреймов, указаны типы знаний, которые они отображают, а также примеры фреймов данного типа из различных предметных областей.

*Таблица 7. Типы фреймов.*

<b>Тип фрейма</b>	<b>Тип знания</b>	<b>Описание</b>	<b>Пример</b>
<b><i>По познавательному назначению</i></b>			
Фреймы-прототипы (шаблоны, образцы)	интенциональные	отражают знания об абстрактных стереотипных понятиях, которые являются классами каких-то конкретных объектов	человек, автомобиль
Фреймы-экземпляры (примеры)	экстенциональные	отражают знания о конкретных фактах предметной области	Иванов И.И., ВАЗ-2110
<b><i>По функциональному назначению</i></b>			
Фреймы-	деклара-	отображают абстрактные и	заем, залог,

структуры (объекты)	типичные	конкретные предметы и понятия предметной области (содержат набор характеристик, описывающий объект или понятие)	вексель, человек, лекция
Фреймы-операции	процедурные	отображают различные процессы преобразования или использования объектов предметной области (содержат набор характеристик процесса)	процессы получения заёма, синтеза устройств
Фреймы-ситуации	прагматические	отображают типичные ситуации, в которых могут находиться фреймы объекты и фреймы роли (содержат набор характеристик, идентифицирующих ситуацию)	авария, тревога, рабочий режим устройства
Фреймы-сценарии	технологические	отображают развитие ситуации, типовую структуру для некоторого действия, понятия, события, отображает динамику (содержат набор характеристик, позволяющих обеспечить развитие системы по данному сценарию)	банкротство, празднование именин, сдача экзамена
Фреймы-роли	функциональные	отображают типичную роль, выполняемую фреймом-объектом в определенной ситуации (содержат набор характеристик роли)	менеджер, кассир, клиент, студент, преподаватель

### *Пример решения задачи*

**Задача.** Построить фреймовую модель представления знаний в предметной области «Ресторан» (посещение ресторана).

**Описание процесса решения.** Для построения фреймовой модели представления знаний необходимо выполнить следующие шаги:

1) Определить абстрактные объекты и понятия предметной области, необходимые для решения поставленной задачи. Оформить их в виде фреймов-прототипов (фреймов-объектов, фреймов-ролей).

2) Задать конкретные объекты предметной области. Оформить их в виде фреймов-экземпляров (фреймов-объектов, фреймов-ролей).

3) Определить набор возможных ситуаций. Оформить их в виде фреймов-ситуаций (прототипы). Если существуют прецеденты по ситуациям в предметной области, добавить фреймы-экземпляры (фреймы-ситуации).

4) Описать динамику развития ситуаций (переход от одних к другим) через набор сцен. Оформить их в виде фреймов-сценариев.

5) Добавить фреймы-объекты сценариев и сцен, которые отражают данные конкретной задачи.

### **Решение.**

1) Ключевые понятия данной предметной области – ресторан, тот, кто посещает ресторан (клиент) и те, кто его обслуживают (повара, метрдотели, официанты, для простоты ограничимся только официантами). У обслуживающего персонала и клиентов есть общие характеристики, поэтому целесообразно выделить общее абстрактное понятие – человек. Тогда фреймы «Ресторан» и «Человек» являются прототипами-образцами, а фреймы «Официант» и «Клиент» - прототипами-ролями. Также нужно определить основные слоты фреймов – характеристики, имеющие значения для решаемой задачи.

<b>ЧЕЛОВЕК</b>			
<b>Имя слота</b>	<b>Значение слота</b>	<b>Способ получения значения</b>	<b>Демон</b>
пол	Мужской или	из внешних источников	
возраст	От 0 до 120 лет	из внешних источников	

<b>РЕСТОРАН</b>			
<b>Имя слота</b>	<b>Значение слота</b>	<b>Способ получения значения</b>	<b>Демон</b>
Название		из внешних источников	
Адрес		из внешних источников	
Часы работы		из внешних источников	
Специализация		из внешних источников	
Класс	Средний или высший	из внешних источников	

Фреймы-наследники содержат все слоты своих родителей, они явно прописываются только в случае изменения какого-либо параметра.

<b>ОФИЦИАНТ (АКО ЧЕЛОВЕК)</b>			
<b>Имя слота</b>	<b>Значение слота</b>	<b>Способ получения значения</b>	<b>Демон</b>
возраст	От 18 до 55 лет	из внешних источников	
стаж работы		из внешних источников	
зарплата		из внешних источников	
график работы		из внешних источников	
место работы	Фрейм-объект	из внешних источников	

<b>КЛИЕНТ (АКО ЧЕЛОВЕК)</b>			
<b>Имя слота</b>	<b>Значение слота</b>	<b>Способ получения значения</b>	<b>Демон</b>
Вид оплаты	Наличные или карточка	По умолчанию (наличные)	
Статус	Обычный или Vip	По умолчанию (обычный)	
Форма заказа	Заказ есть или нет	По умолчанию (заказа нет)	
Чаевые		Из внешних источников	

2) Фреймы-образцы описывают конкретную ситуацию: какие рестораны имеются в городе, как именно организовывается посещение, кто является посетителем, кто работает в выбранном ресторане и т.д. Поэтому определим следующие фреймы-образцы, являющиеся наследниками фреймов-прототипов:



<b>КАФЕ-РЕСТОРАН "ВКУСНЯТИНА" (АКО РЕСТОРАН)</b>			
<b>Имя слота</b>	<b>Значение слота</b>	<b>Способ получения значения</b>	<b>Демон</b>
Название	Вкуснятина	из внешних источников	
Адрес	г. Ульяновск, улица Минаева, 15	из внешних источников	
Часы работы	9:00-00:00	из внешних источников	
Специализация	Пиццерия	из внешних источников	
Класс	Средний или высший	из внешних источников	

<b>КАФЕ "ВКУСНАЯ ЕДА" (АКО РЕСТОРАН)</b>			
<b>Имя слота</b>	<b>Значение слота</b>	<b>Способ получения значения</b>	<b>Демон</b>
Название	Вкусная еда	из внешних источников	
Адрес	г. Ульяновск, улица Карла Маркса, 5	из внешних источников	
Часы работы	9:00-00:00	из внешних источников	
Специализация	Паб	из внешних источников	
Класс	Средний	из внешних источников	

<b>СЕРГЕЙ (АКО ОФИЦИАНТ)</b>			
<b>Имя слота</b>	<b>Значение слота</b>	<b>Способ получения значения</b>	<b>Демон</b>
возраст	27	из внешних источников	
пол	мужской	из внешних источников	
стаж работы	5	из внешних источников	
зарплата	7 000	из внешних источников	
график работы	Через день с 18:00 до 00:00	из внешних источников	
место работы	КАФЕ "ВКУСНАЯ	из внешних источников	

<b>МАРИНА (АКО ОФИЦИАНТ)</b>			
<b>Имя слота</b>	<b>Значение слота</b>	<b>Способ получения значения</b>	<b>Демон</b>
возраст	24	из внешних источников	
Пол	женский	из внешних источников	
стаж работы	2	из внешних источников	
зарплата	8 200	из внешних источников	
график работы	Каждый день с 9:00 до 14:00	из внешних источников	
место работы	КАФЕ-РЕСТОРАН "ВКУСНЯТИНА"	из внешних источников	

<b>ПЁТР (АКО КЛИЕНТ)</b>			
<b>Имя слота</b>	<b>Значение слота</b>	<b>Способ получения значения</b>	<b>Демон</b>
пол	мужской	из внешних источников	
возраст	19	из внешних источников	
Вид оплаты	Наличные	По умолчанию (наличные)	
Статус	Обычный	По умолчанию (обычный)	
Форма заказа	Заказа нет	По умолчанию (заказа нет)	
Чаевые	7 % от суммы заказа	Из внешних источников	

3) Фреймы-ситуации описывают возможные ситуации. В ресторане клиент попадает в несколько типичных ситуаций: заказ и оплата. Возможны и другие не типичные ситуации: клиент подавился, у клиента нет наличности для оплаты счета и т.д. Рассмотрим типичные ситуации (их может быть больше):

<b>ЗАКАЗ</b>			
<b>Имя слота</b>	<b>Значение слота</b>	<b>Способ получения значения</b>	<b>Демон</b>
Перечень блюд		из внешних источников	IF-ADDED (изменяет слот «Перечень цен»)
Перечень цен		Присоединенная процедура	IF-ADDED (изменяет слот «Сумма заказчик»)
Сумма заказа		Присоединенная	
Принял заказ	Фрейм-образец	из внешнего источника	
Сделал заказ	Фрейм-образец	из внешнего источника	

ОПЛАТА			
Имя слота	Значение слота	Способ получения значения	Демон
Вид платежа		из внешних источников	IF-ADDED (изменяет слот «Чаевые»)
Чаевые		Присоединенная	
Оплатил	Фрейм-образец	Присоединенная процедура	
Заказ	Фрейм-образец	из внешних источников	IF-ADDED (изменяет слот «Оплатил»)

4) Ситуации возникают после наступления каких-то событий, выполнения условий и могут следовать одна за другой. Динамику предметной области можно отобразить в фреймах-сценариях. Их может быть множество, опишем наиболее общий и типичный сценарий посещения ресторана:

ПОСЕЩЕНИЕ РЕСТОРАНА			
Имя слота	Значение слота	Способ получения значения	Демон
Посетитель	Фрейм-объект	из внешних источников	
Ресторан	Фрейм-объект	из внешних источников	IF-ADDED, IF-REMOVED (изменяют слот «Официант»)
Официант	Фрейм-объект	присоединенная процедура (определяет по выбранному ресторану)	
Сцена 1	Вход, выбор	из внешних источников	
Сцена 2	Заказ	из внешних источников	
Сцена 3	Еда	из внешних источников	
Сцена 4	Оплата	из внешних источников	
Сцена 5	Выход	из внешних источников	

5) Пусть в рамках нашей задачи Пётр посетил ресторан «Вкусная еда». Тогда фреймы будут заполнены следующим образом:

<b>ПОСЕЩЕНИЕ «Вкусной еды» (АКО ПОСЕЩЕНИЕ РЕСТОРАНА)</b>			
<b>Имя слота</b>	<b>Значение слота</b>	<b>Способ получения значения</b>	<b>Демон</b>
Посетитель	<b>ПЕТР</b>	из внешних источников	
Ресторан	<b>КАФЕ "ВКУСНАЯ ЕДА"</b>	из внешних источников	IF-ADDED, IF-REMOVED (изменяют слот «Официант»)
Официант	<b>СЕРГЕЙ</b>	присоединенная процедура (определяет по выбранному ресторану)	
Сцена 1	Вход, выбор	из внешних источников	
Сцена 2	<b>ЗАКАЗ ПЕТРА</b>	из внешних источников	
Сцена 3	Еда	из внешних источников	
Сцена 4	<b>ОПЛАТА ПЕТРА</b>	из внешних источников	
Сцена 5	Выход	из внешних источников	

<b>ЗАКАЗ ПЕТРА (АКО ЗАКАЗ)</b>			
<b>Имя слота</b>	<b>Значение слота</b>	<b>Способ получения значения</b>	<b>Демон</b>
Перечень блюд	Отбивная, темное пиво	из внешних источников	IF-ADDED (изменяет слот «Перечень цен»)
Перечень цен	250, 75	Присоединенная процедура	IF-ADDED (изменяет слот «Сумма заказк»)
Сумма заказа	325	Присоединенная	
Принял заказ	<b>СЕРГЕЙ</b>	из внешнего источника	
Сделал заказ	<b>ПЕТР</b>	из внешнего источника	

<b>ОПЛАТА ПЕТРА (АКО ОПЛАТА)</b>			
<b>Имя слота</b>	<b>Значение слота</b>	<b>Способ получения значения</b>	<b>Демон</b>
Вид платежа	Наличные	из внешних источников	IF-ADDED (изменяет слот «Чаевые»)
Чаевые	30	Присоединенная процедура	
Оплатил	<b>ПЕТР</b>	из внешних источников	
Заказ	<b>ЗАКАЗ ПЕТРА</b>	из внешних источников	IF-ADDED (изменяет слот «Оплатил»)

Взаимосвязь различных видов фреймов отображается графически в виде графа (рис. 4).

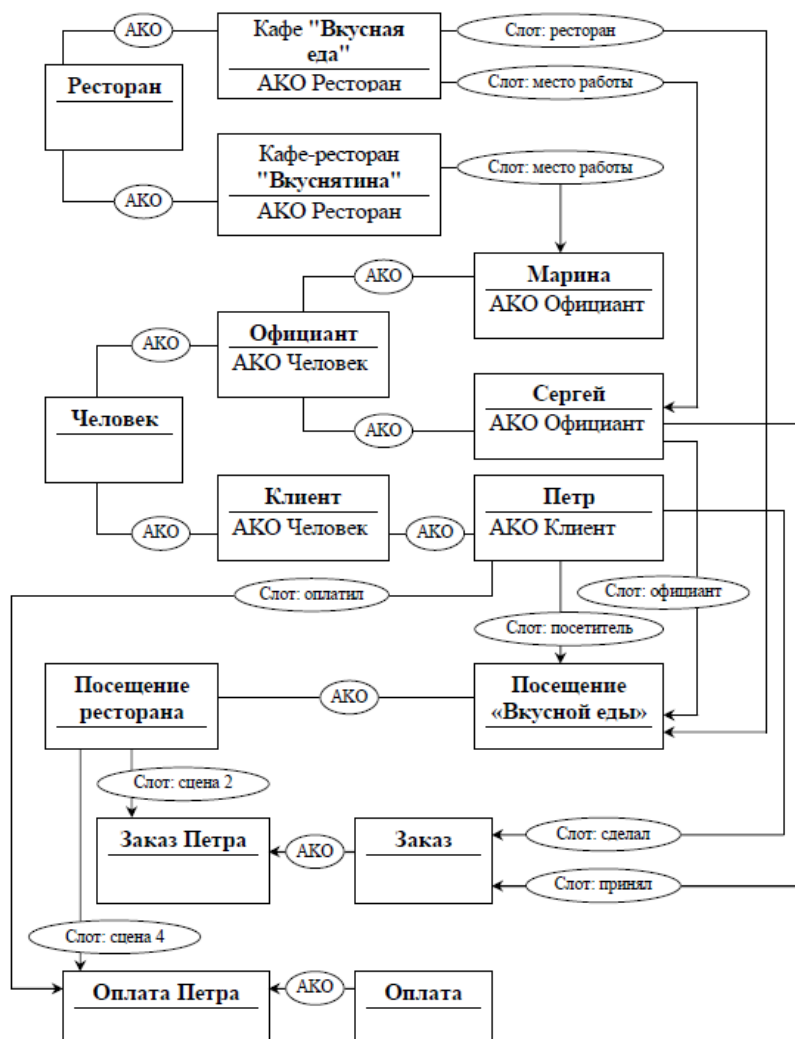


Рис. 4. Схема фреймов для предметной области «Ресторан».

Использование фреймовой модели аналогично семантической, только в процессе получения ответа кроме вершин учитываются и слоты. Например, получить ответ на вопрос «Кто работает официантом в ресторане “Вкусная еда”?» можно следующим образом: из запроса понятно, что необходимо

найти фрейм «Ресторан “Вкусная еда”» и проследить связь с фреймом «Сергей», являющимся наследником фрейма «Официант». Также можно найти слот «Место работы» и проверив его значение во фреймах наследниках фрейма «Официант» определить, что официантом в ресторане “Вкусная еда” работает Сергей.

### *Варианты заданий*

1. Построить производственную модель представления знаний в предметной области «Аэропорт» (диспетчерская).
2. Построить производственную модель представления знаний в предметной области «Железная дорога» (продажа билетов).
3. Построить производственную модель представления знаний в предметной области «Торговый центр» (организация).
4. Построить производственную модель представления знаний в предметной области «Автозаправка» (обслуживание клиентов).
5. Построить производственную модель представления знаний в предметной области «Автопарк» (пассажирские перевозки).
6. Построить производственную модель представления знаний в предметной области «Компьютерные сети» (организация).
7. Построить производственную модель представления знаний в предметной области «Университет» (учебный процесс).
8. Построить производственную модель представления знаний в предметной области «Компьютерная безопасность» (средства и способы ее обеспечения).
9. Построить производственную модель представления знаний в предметной области «Компьютерная безопасность» (угрозы).
10. Построить производственную модель представления знаний в предметной области «Интернет-кафе» (организация и обслуживание).
11. Построить производственную модель представления знаний в предметной области «Разработка информационных систем» (ведение информационного проекта).
12. Построить производственную модель представления знаний в предметной области «Туристическое агентство» (работа с клиентами).
13. Построить производственную модель представления знаний в предметной области «Зоопарк» (организация).

14. Построить продукционную модель представления знаний в предметной области «Кухня» (приготовление пищи).
15. Построить продукционную модель представления знаний в предметной области «Больница» (прием больных).
16. Построить продукционную модель представления знаний в предметной области «Кинопрокат» (ассортимент и работа с клиентами).
17. Построить продукционную модель представления знаний в предметной области «Прокат автомобилей» (ассортимент и работа с клиентами).
18. Построить продукционную модель представления знаний в предметной области «Операционные системы» (функционирование).
19. Построить продукционную модель представления знаний в предметной области «Информационные системы» (виды и функционирование).
20. Построить продукционную модель представления знаний в предметной области «Предприятие» (структура и функционирование).

#### **Лабораторная работа 4 «Элементы логики высказываний. Способы доказательства и вывода»**

**Высказыванием** называется повествовательное предложение, о котором имеет смысл говорить, что оно истинно или ложно.

*Примеры высказываний:* « $2 \times 2 = 4$ », «Волга впадает в Черное море», «Москва – столица России». Первое и третье высказывания истинны, второе – ложно. Предложение « $x + y = 4$ » не является высказыванием, т.к. оно может быть истинным при некоторых значениях  $x$  и  $y$  и ложным при других значениях. Из простых, атомарных высказываний можно сооружать сложные высказывания. Например, из двух высказываний: «Москва стоит на берегу Невы» и «Санкт-Петербург стоит на берегу Невы», из которых первое ложно, второе истинно, можно соорудить более сложные высказывания: «Москва стоит на берегу Невы или Санкт-Петербург стоит на берегу Невы» (истинное) или «Москва стоит на берегу Невы и Санкт-Петербург стоит на берегу Невы» (ложное).

В логике высказываний простые высказывания являются переменными, принимающими значения «истина» (и) или «ложь» (л).

Переменной (и) соответствует 1, переменной (л) – 0. Для них стандартным образом определяются функции: дизъюнкция высказываний, конъюнкция (два последних примера), отрицание, эквивалентность, неравнозначность (исключающее «или»), импликация.

- *Конъюнкцией* (операция «И») двух высказываний называется высказывание, истинное, когда оба высказывания истинны, и ложное – в других случаях. Обозначается:  $P \wedge Q$  или  $P \cdot Q$ .
- *Дизъюнкцией* (операция «ИЛИ») двух высказываний называется высказывание, ложное в случае, когда оба высказывания ложны, и истинное – в других случаях. Обозначается:  $P \vee Q$ .
- *Отрицанием* (операция «НЕ») высказывания  $P$  называется высказывание, истинное, когда  $P$  ложно, и ложное – в противном случае. Обозначается:  $\bar{P}$ .
- *Импликацией* (логическим следованием) двух высказываний называется высказывание, ложное, когда первое истинно, а второе ложно; во всех других случаях – истинное. Обозначается:  $P \rightarrow Q$ ,  $P$  – посылка,  $Q$  – заключение.
- *Эквиваленцией* (равнозначностью) двух высказываний называется высказывание, истинное, когда значения исходных высказываний совпадают, ложное – в противном случае. Обозначается:  $P \leftrightarrow Q$  или  $P \sim Q$ .
- *Неравнозначностью* (операция «исключающее ИЛИ») высказываний называется высказывание, истинное, когда значения исходных высказываний не совпадают, ложное – в противном случае. Обозначается:  $P \oplus Q$ .

Простые высказывания будем обозначать буквами. Буквы, обозначающие высказывания, логические связи и скобки составляют алфавит. С помощью элементов алфавита можно строить разнообразные **логические формулы**.

В логике изучается строение сложных высказываний, выраженных формулами, вне зависимости от содержания составляющих их простых высказываний.

Иногда значение конкретной логической формулы не зависит от значений входящих в них переменных. Правильно построенные логические



формулы, значением которых будет «истина» при любых значениях входящих в них переменных, называются **тавтологиями**.

В противовес тавтологиям в логике существуют **противоречия** – формулы, значением которых всегда будет «ложь» независимо от значений входящих в них переменных.

Наряду с алфавитом и правилами построения сложных высказываний - логических формул, язык логики высказываний содержит **правила преобразования логических формул**, которые реализуют **логически правильные рассуждения**.

Процесс получения новых знаний, выраженных высказываниями, из других знаний называется **рассуждением**. Исходные высказывания называются **посылками** (гипотезами, условиями), а получаемые – **заключением** (следствием).

Основной проблемой логики является **проблема доказательства**, которая заключается в решении вопроса, можно ли некоторую цепь рассуждений, основываясь только на ее структуре, считать правильной. Т.е. в нахождении истинного значения  $B$ , если предполагается истинность исходных посылок  $A_1, A_2, \dots, A_N$ . Например, следует ли из высказывания «если человек думает как я, то он хороший человек» заключение «если человек думает не как я, то он плохой»?

**Рассуждение называется правильным**, если из конъюнкции посылок  $A_1, A_2, \dots, A_N$  следует заключение  $B$ , это записывается:

$$\frac{A_1, A_2, \dots, A_N}{B}$$

т.е. всякий раз, когда все посылки истинны, то заключение тоже истинно.

Таким образом, чтобы установить правильность рассуждений, надо показать, что формула  $(A_1 \wedge A_2 \wedge \dots \wedge A_N) \rightarrow B$  является тавтологией, т.е.  $(A_1 \wedge A_2 \wedge \dots \wedge A_N) \rightarrow B \equiv 1$ . Чтобы установить ложность рассуждений, надо показать, что данная формула тавтологией не является.

Существуют различные методы решения проблемы доказательства в логике. Рассмотрим некоторые из них.

- **Метод построения таблицы истинности** состоит в следующем. Можно перечислить все переменные, входящие в формулы  $A_1, A_2, \dots, A_N, B$ , и составить таблицу истинности для всевозможных

комбинаций значений этих переменных. Затем следует осуществить просмотр полученной таблицы, чтобы проверить, во всех ли ее строках, где формулы  $A_1, A_2, \dots, A_N$  имеют значения «истина», формула  $B$  также имеет значение «истина». Этот метод применим всегда, но может оказаться слишком трудоемким.

- При доказательстве *методом преобразований* сначала записывают формулу  $(A_1 \wedge A_2 \wedge \dots \wedge A_N) \rightarrow B$  и, применяя к ней законы алгебры логики, стараются упростить выражение. Этот процесс продолжают до тех пор, пока не будет получено выражение, равное значению «истина», либо выражение, про которое однозначно можно сказать, что оно тождественно истинным не является.

Таблица 8. Законы алгебры логики.

Закон	для конъюнкции	для дизъюнкции
Закон тождества	$1A=A$	$1 \vee A=1$
Закон нуля	$0A=0$	$0 \vee A=A$
Закон идемпотентности	$AA=A$	$A \vee A=A$
Закон инверсии	$A \bar{A}=0$	$A \vee \bar{A}=1$
Коммутативный закон	$AB=BA$	$A \vee B=B \vee A$
Ассоциативный закон	$(AB)C=A(BC)$	$(A \vee B) \vee C=A \vee (B \vee C)$
Дистрибутивный закон	$A(B \vee C)=AB \vee AC$	$A \vee (B \wedge C)=(A \vee B)(A \vee C)$
Закон поглощения	$A(A \vee B)=A$	$A \vee (A \wedge B)=A$
Закон де Моргана	$\overline{(A \wedge B)}=\bar{A} \vee \bar{B}$	$\overline{(A \vee B)}=\bar{A} \wedge \bar{B}$

Дополнительные соотношения

$$A \rightarrow B = \bar{A} \vee B, \quad A \sim B = AB \vee \bar{A} \bar{B}, \quad A \oplus B = \bar{A} \bar{B} \vee \bar{A} B$$

### Пример решения задачи

**Задача.** Выяснить, являются ли следующие рассуждения логически верными.

Если Джонс не встречал ночью Смита, то Смит был убийцей или Джонс лжет. Если Смит не был убийцей, то Джонс не встречал Смита этой ночью, и убийство имело место после полуночи. Если убийство имело место после полуночи, то Смит был убийцей или Джонс не лжет. Следовательно, Смит был убийцей.

**Описание процесса решения.** Для проверки приведенных рассуждений на логическую правильность необходимо:

1) Ввести логические переменные, обозначив ими простые высказывания.

2) Записать сложные высказывания в виде логических формул, используя введенные переменные и известные логические связи.

3) Выделить в структуре рассуждений посылки  $A_1, A_2, \dots, A_N$  и заключение  $B$ .

4) Составить формулу  $(A_1 \wedge A_2 \wedge \dots \wedge A_N) \rightarrow B$  и проверить, является ли она тавтологией:

а) Метод преобразований. С помощью законов алгебры логики показать, что  $(A_1 \wedge A_2 \wedge \dots \wedge A_N) \rightarrow B \equiv 1$ .

б) Метод построения таблицы истинности. С помощью построения таблицы истинности показать, что  $(A_1 \wedge A_2 \wedge \dots \wedge A_N) \rightarrow B \equiv 1$ .

### **Решение.**

1) Введем логические переменные:

$x$  – «Джонс не встречал ночью Смита»,

$y$  – «Смит убийца»,

$z$  – «Джонс лжет»,

$t$  – «убийство состоялось после полуночи».

2) Прежде чем записать формулу, надо уточнить по условию задачи в каком контексте употребляется союз «или». Когда мы говорим « $A$  или  $B$ », мы можем подразумевать две разные ситуации: а)  $A \vee B$  или б)  $A\bar{B} \vee \bar{A}B$ . Во втором случае высказывания  $A$  и  $B$  не могут быть одновременно истинными. Чтобы подчеркнуть этот момент, обычно говорят «либо  $A$ , либо  $B$ ». В нашей задаче нет такой оговорки, поэтому мы можем для записи высказывания: «Смит был убийцей или Джонс не лжет» использовать формулу  $y \vee \bar{z}$ .

3) Итак, мы имеем посылки:  $x \rightarrow (y \vee z)$ ,  $\bar{y} \rightarrow xt$ ,  $t \rightarrow (y \vee \bar{z})$ , заключение:  $y$ .

Надо составить формулу:  $(x \rightarrow (y \vee z)) \cdot (\bar{y} \rightarrow xt) \cdot (t \rightarrow (y \vee \bar{z})) \rightarrow y$

и посмотреть, будет ли она тавтологией.

4) а) **Метод преобразований.**

Применяя законы алгебры логики и дополнительные соотношения, упростим формулу и получим:

$$\begin{aligned} & (x \rightarrow (y \vee z)) \cdot (\bar{y} \rightarrow xt) \cdot (t \rightarrow (y \vee \bar{z})) \rightarrow y = \\ & = (\bar{x} \vee y \vee z) \cdot (y \vee xt) \cdot (\bar{t} \vee y \vee \bar{z}) \rightarrow y = \\ & = (y \vee (\bar{x} \vee z)) \cdot xt \cdot (\bar{t} \vee \bar{z}) \rightarrow y = (y \vee xzt \cdot (\bar{t} \vee \bar{z})) \rightarrow y = y \rightarrow y \equiv 1. \end{aligned}$$

Полученное выражение является тавтологией (т.е. тождественно равно 1). Следовательно, рассуждения логически правильны.

б) **Метод построения таблицы истинности.**

Расчеты будем проводить в таблице. Т.к. имеем четыре логические переменные, то возможных комбинаций значений этих переменных будет  $2 \cdot 2 \cdot 2 \cdot 2 = 16$ . Пошагово вычислим значения входящих в формулу  $(x \rightarrow (y \vee z)) \cdot (\bar{y} \rightarrow xt) \cdot (t \rightarrow (y \vee \bar{z})) \rightarrow y$  выражений, используя определения конъюнкции, дизъюнкции, отрицания и импликации.

Значению «истина» соответствует 1, значению «ложь» – 0.

$x$	$y$	$z$	$t$	$y \vee z$	$x \rightarrow (y \vee z)$ $A$	$xt$	$\bar{y} \rightarrow xt$ $B$	$y \vee \bar{z}$	$t \rightarrow (y \vee \bar{z})$ $C$	$ABC$	$ABC \rightarrow y$
0	0	0	0	0	1	0	0	1	1	0	1
0	0	0	1	0	1	0	0	1	1	0	1
0	0	1	0	1	0	0	0	0	1	0	1
0	0	1	1	1	0	0	0	0	0	0	1
0	1	0	0	1	1	0	1	1	1	1	1
0	1	0	1	1	1	0	1	1	1	1	1
0	1	1	0	1	0	0	1	1	1	0	1
0	1	1	1	1	0	0	1	1	1	0	1
1	0	0	0	0	1	0	0	1	1	0	1
1	0	0	1	0	0	1	1	1	1	0	1
1	0	1	0	1	1	0	0	0	1	0	1
1	0	1	1	1	1	1	1	0	0	0	1
1	1	0	0	1	1	0	1	1	1	1	1
1	1	0	1	1	1	1	1	1	1	1	1
1	1	1	0	1	1	0	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1

Как видим, значение формулы  $(x \rightarrow (y \vee z)) \cdot (\bar{y} \rightarrow xt) \cdot (t \rightarrow (y \vee \bar{z})) \rightarrow y$  (для компактности в таблице она обозначена как  $ABC \rightarrow y$ ) на всевозможных наборах входящих переменных равна 1. Формула является тавтологией. Следовательно, рассуждения логически правильны.

### *Варианты заданий*

1. Профсоюзы штата будут поддерживать губернатора, если он подпишет этот закон. Фермеры окажут ему поддержку, если он наложит на него вето. Очевидно, что он или не подпишет закон, или не наложит на него вето. Следовательно, губернатор потеряет голоса рабочих, объединенных в профсоюзы, или голоса фермеров.
2. Если мы не будем продолжать политику сохранения цен, то мы потеряем голоса фермеров. Если же мы будем продолжать эту политику и не прибегнем к контролю над производством, то продолжится перепроизводство. Без голосов фермеров нас не переизберут. Значит, если нас переизберут, и мы не прибегнем к контролю над производством, то продолжится перепроизводство.
3. Если завтра будет хорошая погода, то я буду кататься на коньках или я пойду на лыжах. Если я пойду на лыжах, то лучше поехать за город, а если буду кататься на коньках, то останусь в городе. Мне не хочется завтра в выходной день оставаться в городе. Следовательно, если завтра будет хорошая погода, то я пойду на лыжах.
4. В бюджете возникнет дефицит, если не повысят пошлины. Если в бюджете возникнет дефицит, то расходы на социальные нужды сократятся. Следовательно, если повысят пошлины, то расходы на социальные нужды не сократятся.
5. Намеченная атака удастся, если захватить противника врасплох или его позиции плохо защищены. Захватить противника врасплох можно только, если он беспечен. Он не будет беспечен, если его позиции плохо защищены. Следовательно, намеченная атака не удастся.
6. Если губернатор не имеет соответствующего авторитета или если он не желает принимать на себя ответственность, то порядок не будет восстановлен и волнения не прекратятся до тех пор, пока участникам волнений это не надоест, и власти не начнут примирительные действия. Следовательно, если губернатор не желает взять на себя

ответственность и участникам волнений это не надоест, то волнения не прекратятся.

7. Если налоги в бюджет не собраны, то либо секвестрируется бюджет, либо правительство уходит в отставку. Если секвестрируется бюджет, то падает уровень жизни. Налоги в бюджет не собраны. Следовательно, либо падает уровень жизни, либо уровень жизни не падает и правительство уходит в отставку.
8. Экзамен сдан вовремя или сессия продлена. Если сессия продлена, то не сдана курсовая работа или не зачтены лабораторные работы. Курсовая работа сдана. Экзамен вовремя не сдан. Следовательно, неверно, что если курсовая работа сдана, то лабораторные работы зачтены.
9. Если имеет место денежная эмиссия, то растет курс доллара. Если эмиссии нет и инфляция не растет, то курс доллара не растет. Инфляция не растет. Следовательно, имеет место эмиссия и растет курс доллара или нет эмиссии и курс доллара не растет.
10. Зарботная плата возрастет только, если будет инфляция. Если будет инфляция, то увеличится стоимость жизни. Зарботная плата возрастет. Следовательно, стоимость жизни увеличится.
11. Если 2 - простое число, то это наименьшее простое число. Если 2 - наименьшее простое число, то 1 не есть простое число. Число 1 не есть простое число. Следовательно, 2 - простое число.
12. Если идет дождь, то нежарко. Если светит солнце, то жарко. Идет дождь. Следовательно, нежарко и не светит солнце.
13. Если завтра будет холодно, я надену теплое пальто, если рукав будет починен. Завтра будет холодно, а рукав не будет починен. Следовательно, я не надену теплое пальто.
14. Если исход скачек будет предрешен сговором или в игорных домах будут орудовать шулеры, то доходы от туризма упадут и город пострадает. Если доходы от туризма упадут, полиция будет довольна. Полиция никогда не бывает довольна. Следовательно, исход скачек не предрешен сговором.
15. Или Сэлли и Боб одного возраста, или Сэлли старше Боба. Если Сэлли и Боб одного возраста, то Нэнси и Боб не одного возраста. Если Сэлли

старше Боба, то Боб старше Уолтера. Следовательно, или Нэнси и Боб не одного возраста, или Боб старше Уолтера.

16. Джон или переутомился или он болен. Если он переутомился, то он раздражается. Он не раздражается. Следовательно, Джон болен.
17. Если строить противоатомные убежища, то другие государства будут чувствовать себя в опасности, а наш народ получит ложное представление о своей безопасности. Если другие страны будут чувствовать себя в опасности, то они смогут начать превентивную войну. Если наш народ получит ложное представление о своей безопасности, то он ослабит свои усилия, направленные на сохранение мира. Если же не строить противоатомные убежища, то мы рискуем иметь колоссальные потери в случае войны. Следовательно, либо другие страны могут начать превентивную войну, и наш народ ослабит свои усилия, направленные на сохранение мира, либо мы рискуем иметь колоссальные потери в случае войны.

## **Лабораторная работа 5 «Элементы нечеткой логики. Нечеткий вывод»**

*Нечетким множеством*  $A$  называется множество пар  $A = \{(x, \mu_A(x)); x \in X\}$ , где  $\mu_A(x)$  – степень (функция) принадлежности нечеткого множества  $A$ . Эта функция приписывает каждому элементу  $x \in X$  степень его принадлежности к нечеткому множеству  $A$ , при этом можно выделить три случая:

1.  $\mu_A(x) = 1$  означает полную принадлежность элемента  $x$  к нечеткому множеству  $A$ , т.е.  $x \in A$ ;
2.  $\mu_A(x) = 0$  означает отсутствие принадлежности элемента  $x$  к нечеткому множеству  $A$ , т.е.  $x \notin A$ ;
3.  $0 < \mu_A(x) < 1$  означает частичную принадлежность элемента  $x$  к нечеткому множеству  $A$ .

*Пример.*

$$U = \{a, b, c, d, e\},$$

$$A = \{(a, 0), (b, 0, 1), (c, 0, 5), (d, 0, 9), (e, 1)\}$$

1. элемент  $a$  не принадлежит множеству  $A$ ;
2. элемент  $b$  принадлежит множеству  $A$  в малой степени;

3. элемент  $s$  принадлежит множеству  $A$  более или и менее;
4. элемент  $d$  принадлежит множеству  $A$  в значительной степени;
5. элемент  $e$  принадлежит множеству  $A$ .

В нечеткой логике вводится понятие *лингвистической переменной*, значениями которой являются не числа, а слова естественного языка, называемые термами. Например, лингвистическая переменная «возраст» может иметь значения «юный», «молодой», «пожилой» и т.д. (рис.5). Значения лингвистической переменной определяются через нечеткие множества, которые, в свою очередь, определены на некотором базовом наборе значений  $X$  или базовой числовой шкале, имеющей размерность.

*Лингвистической переменной* называется набор  $\langle \beta, T, X, G, M \rangle$ , где  $\beta$  – наименование лингвистической переменной,  $T$  – множество ее значений (терм-множество), представляющих собой наименования нечетких переменных, областью определения каждой из которых является множество  $X$  (множество  $T$  называется базовым терм-множеством лингвистической переменной),  $G$  – синтаксическая процедура, позволяющая оперировать элементами терм-множества  $T$ , в частности, генерировать новые термы (значения),  $M$  – семантическая процедура, позволяющая превратить каждое новое значение лингвистической переменной, образуемое процедурой  $G$ , в нечеткую переменную, т.е. сформировать соответствующее нечеткое множество.

*Нечеткая переменная* характеризуется тройкой  $\langle \alpha, X, A \rangle$ , где  $\alpha$  – наименование переменной,  $X$  – универсальное множество (область определения  $\alpha$ ),  $A$  – нечеткое множество на  $X$ , описывающее ограничения на значения нечеткой переменной  $\alpha$ .

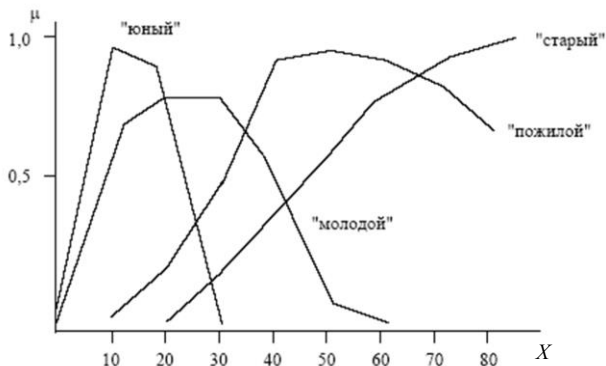




Рис. 5. Пример лингвистической переменной «Возраст».

**Нечетким высказыванием** называется повествовательное предложение, выражающее законченную мысль, относительно которой мы можем судить об ее истинности или ложности только с некоторой степенью уверенности.

Над нечеткими высказываниями определены следующие основные операции. Пусть заданы два нечетких высказывания  $A$  и  $B$ , где  $\mu(A)$  и  $\mu(B)$  – степени уверенности (истинности) соответственно.

- *Конъюнкцией* нечетких высказываний  $A$  и  $B$  называется нечеткое высказывание  $A \wedge B$ , истинность которого принимает значение  $\mu(A \wedge B) = \min(\mu(A), \mu(B))$ .
- *Дизъюнкцией* нечетких высказываний  $A$  и  $B$  называется нечеткое высказывание  $A \vee B$ , истинность которого принимает значение  $\mu(A \vee B) = \max(\mu(A), \mu(B))$ .
- *Отрицанием* нечеткого высказывания  $A$  называется нечеткое высказывание  $\bar{A}$ , истинность которого принимает значение  $\mu(\bar{A}) = 1 - \mu(A)$ .
- *Импликацией* нечетких высказываний называется  $A \rightarrow B$  называется нечеткое высказывание  $A \rightarrow B$  (читается - "из  $A$  следует  $B$ ", "ЕСЛИ  $A$ , ТО  $B$ "), истинность которого принимает значение  $\mu(A \rightarrow B) = \max\{1 - \mu(A), \mu(B)\}$ .

Нечеткая импликация занимает центральное место в системах нечеткого вывода. *Продукционная нечеткая система* или система нечетких правил продукций представляет собой некоторое согласованное множество отдельных нечетких продукций в форме "ЕСЛИ  $A$ , ТО  $B$ ". Основная задача приближенных рассуждений с использованием нечетких правил продукций – на основе некоторых нечетких высказываний с известной степенью истинности, которые являются условиями нечетких правил продукций, оценить степень истинности других нечетких высказываний, являющимися заключениями соответствующих нечетких правил продукций.

Основными этапами разработки и применения систем нечёткого логического вывода являются следующие.

1) *Этап формирования правил* – формализация экспертом базы знаний системы нечёткого логического вывода в виде нечётких продукционных правил заданного вида.

2) *Фаззификация* (введение нечеткости) входных переменных – установление соответствия между конкретным (обычно численным) значением отдельной входной переменной системы нечеткого вывода и значением функции принадлежности соответствующего ей терма водной лингвистической переменной.

3) *Агрегирование подусловий* – оценка степени истинности условий правил базы знаний по каждому из правил системы.

4) *Активизация подзаключений* – оценка истинности отдельных заключений правил базы знаний (в нечёткой форме) на основе оценок истинности нечётких условий данных правил.

5) *Аккумуляция заключений* – формирование нечёткого заключения для всей системы нечётких продукций базы знаний. Данное заключение определяет нечёткий выход системы нечёткого логического вывода.

6) *Дефаззификация* – приведение к чёткости, требуемое при необходимости преобразования нечёткого результата в чёткое число. Наиболее известными методами дефаззификации являются методы центра тяжести, деления площади пополам, максимума принадлежности.

### *Пример решения задачи*

**Задача.** Построить нечеткую базу знаний (использовать не менее 3 лингвистических переменных) для задачи определения временных затрат для решения студентом задач данного пособия (учитывать успеваемость студента и количество решаемых вариантов), проверить ее на полноту и произвести нечеткий вывод для конкретных значений (выбрать случайным образом).

**Описание процесса решения.** Для построения нечеткой базы знаний и реализации логического вывода необходимо выполнить следующее:

1) Сформулировать на естественном языке в виде предложений «Если..., то...» закономерности предметной области.

2) Выделить из этих предложений лингвистические переменные, их значения (построить их функции принадлежности), высказывания различных видов, формализовать нечеткие правила.

- 3) Проверить полученную базу знаний на полноту.
- 4) Провести фаззификацию (входные данные выбираем случайным образом).
- 5) Провести агрегирование подусловий и активизацию подзаключений.
- 6) Провести аккумуляирование заключений.
- 7) Провести дефаззификацию.

### **Решение.**

1) Предложения, описывающие данную задачу могут быть такими:

• *Если успеваемость студента высокая или хорошая и он прорешивает малое количество вариантов, то ему требуется немного времени.*

• *Если успеваемость студента высокая или хорошая и он прорешивает много вариантов, то ему требуется достаточно большой промежуток времени.*

• *Если успеваемость студента низкая и он прорешивает много вариантов, то ему требуется много времени.*

• *Если успеваемость студента средняя и он прорешивает достаточно большое количество вариантов, то ему требуется достаточно большой промежуток времени.*

2) Выделим из предложений лингвистические переменные.

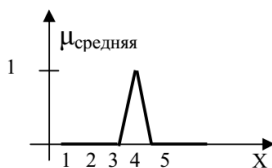
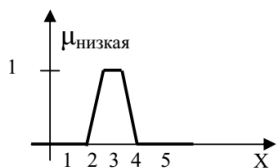
1.  $\beta_1$  (входная) – «Успеваемость студента», терм-множество  $T=(\text{«высокая»}, \text{«средняя»}, \text{«низкая»})$ , базовое множество  $X=[2,5]$  (используется пятибалльная система).

2.  $\beta_2$  (входная) – «Количество вариантов», терм-множество  $T=(\text{«мало»}, \text{«достаточно»}, \text{«много»})$ , базовое множество  $X=[1,20]$  (количество вариантов 20 в каждой теме).

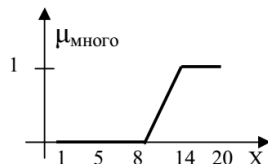
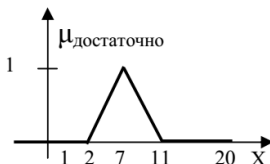
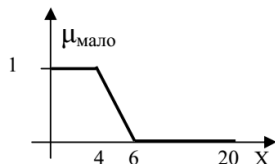
3.  $\beta_3$  (выходная) – «Количество времени», терм-множество  $T=(\text{«мало»}, \text{«достаточно»}, \text{«много»})$ , базовое множество  $X=[1,7]$  (количество часов в неделю, уделенных предмету изучения).

Для полного задания лингвистической переменной необходимо определить нечеткие переменные, входящие в  $T$ :

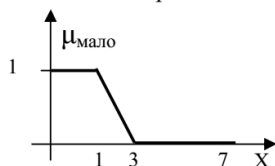
Успеваемость:



Количество вариантов:



Количество времени:



С учетом выделенных лингвистических переменных, нечеткие правила следующие:

1. Если Успеваемость = «высокая» или Успеваемость = «средняя» и Количество вариантов = «мало», то Количество времени = «мало».

2. Если Успеваемость = «высокая» или Успеваемость = «средняя» и Количество вариантов = «много», то Количество времени = «достаточно».

3. Если Успеваемость = «низкая» и Количество вариантов = «много», то Количество времени = «много».

4. Если Успеваемость = «средняя» и Количество вариантов = «достаточно», то Количество времени = «достаточно».

3) Проверим полученную базу на полноту:

- существует хотя бы одно правило для каждого лингвистического термина выходной переменной (выходная переменная «Количество времени» имеет 3 термина: «мало» используется в 1 правиле, «достаточно» - в 2 и 4, «много» - в третьем);
- для любого термина входной переменной имеется хотя бы одно правило, в котором этот термин используется в качестве посылки (есть две входные переменные «Успеваемость» и «Количество вариантов», у каждой из

них 3 термина: «высокая» используется в 1 и 2 правилах, «средняя» - 1, 2 и 4, «низкая» - в 3, «мало» - в 1, «достаточно» - 4, «много» - 3 и 2).

Значит, полученная база нечетких правил полная.

4) Пусть имеется студент Иванов А.А., имеющий среднюю оценку 3,5 и решивший прорешать 9 вариантов, нужно определить сколько ему понадобится времени.

Определим степени уверенности простейших утверждений:

Успеваемость = «высокая» - 0;

Успеваемость = «средняя» - 0.5;

Успеваемость = «низкая» - 1;

Количество вариантов = «мало» - 0;

Количество вариантов = «достаточно» - 0.5;

Количество вариантов = «много» - 0.125.

5) Определим степени уверенности посылок правил:

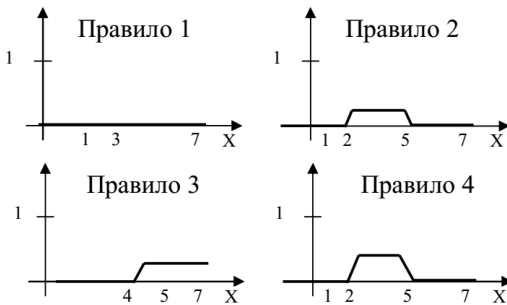
Правило 1:  $\min(\max(0, 0.5), 0) = 0$ ;

Правило 2:  $\min(\max(0, 0.5), 0.125) = 0.125$ ;

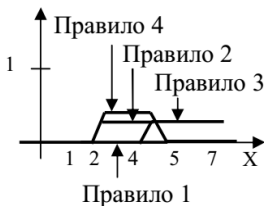
Правило 3:  $\min(1, 0.125) = 0.125$ ;

Правило 4:  $\min(0.5, 0.5) = 0.5$ .

Построим новую выходную нечеткую переменную, используя полученные степени уверенности:



б) Аккумуляция:



Новый терм выходной переменной Количество часов:



7) Исходя из полученного графика степени принадлежности выходного термина, можно сказать, что Иванову А.А., имеющему среднюю оценку 3.5, на решение 9 вариантов заданий понадобится не менее 2.75 часа (степень уверенности данного утверждения 0.5).

### *Варианты заданий*

1. Построить нечеткую базу знаний (использовать не менее 3 лингвистических переменных) для задачи закупок (соотношения цены, качества, объема закупок и т.д.), проверить ее на полноту и произвести нечеткий вывод для конкретных значений (выбрать случайным образом).
2. Построить нечеткую базу знаний (использовать не менее 3 лингвистических переменных) для задачи распределения нагрузок спортсмена (соотношение нагрузок, физического состояния, потребляемых калорий и т.д.), проверить ее на полноту и произвести нечеткий вывод для конкретных значений (выбрать случайным образом).
3. Построить нечеткую базу знаний (использовать не менее 3 лингвистических переменных) для задачи управления транспортным средством (регулировка скорости с учетом передачи, погодных условий, интенсивности потока и т.д.), проверить ее на полноту и произвести нечеткий вывод для конкретных значений (выбрать случайным образом).
4. Построить нечеткую базу знаний (использовать не менее 3 лингвистических переменных) для задачи управления транспортным средством (управление рулем, газом, тормозом при въезде в гараж), проверить ее на полноту и произвести нечеткий вывод для конкретных значений (выбрать случайным образом).
5. Построить нечеткую базу знаний (использовать не менее 3 лингвистических переменных) для задачи регулирования

теплоснабжения (соотношение среднесуточной температуры, ветра, размера здания и т.д.), проверить ее на полноту и произвести нечеткий вывод для конкретных значений (выбрать случайным образом).

6. Построить нечеткую базу знаний (использовать не менее 3 лингвистических переменных) для задачи регулирования реверсного движения на волжском мосту (учитывать время, интенсивность потока, день недели и т.д.), проверить ее на полноту и произвести нечеткий вывод для конкретных значений (выбрать случайным образом).
7. Построить нечеткую базу знаний (использовать не менее 3 лингвистических переменных) для задачи подбора специй для блюда (соотношение количества и остроты специй, рецептуры, предпочтений едока, объема пищи и т.д.), проверить ее на полноту и произвести нечеткий вывод для конкретных значений (выбрать случайным образом).
8. Построить нечеткую базу знаний (использовать не менее 3 лингвистических переменных) для задачи подбора объема блюд (учитывать калорийность, вкусовые предпочтения, количество едоков и т.д.), проверить ее на полноту и произвести нечеткий вывод для конкретных значений (выбрать случайным образом).
9. Построить нечеткую базу знаний (использовать не менее 3 лингвистических переменных) для задачи подачи электроэнергии в условиях экономии (учет времени суток, типа помещений, количества людей, типа оборудования и т.д.), проверить ее на полноту и произвести нечеткий вывод для конкретных значений (выбрать случайным образом).
10. Построить нечеткую базу знаний (использовать не менее 3 лингвистических переменных) для задачи подбора интенсивности занятий (учитывать начальный уровень подготовки, объем учебного материала, количество человек в группе, необходимый уровень усвоения и т.д.), проверить ее на полноту и произвести нечеткий вывод для конкретных значений (выбрать случайным образом).
11. Построить нечеткую базу знаний (использовать не менее 3 лингвистических переменных) для задачи расчета потребления бензина (учитывать тип совершаемых маневров, уровень подготовки водителя, состояние автомобиля, тип автомобиля и т.д.), проверить ее на полноту

и произвести нечеткий вывод для конкретных значений (выбрать случайным образом).

12. Построить нечеткую базу знаний (использовать не менее 3 лингвистических переменных) для задачи регулирования системы орошения (учитывать время года, количество выпадающих озадков, вид орошаемой культуры и т.д.), проверить ее на полноту и произвести нечеткий вывод для конкретных значений (выбрать случайным образом).
13. Построить нечеткую базу знаний (использовать не менее 3 лингвистических переменных) для задачи настройки аудиосистемы (мощность колонок, их количество, размер помещения, назначение установки и т.д.), проверить ее на полноту и произвести нечеткий вывод для конкретных значений (выбрать случайным образом).
14. Построить нечеткую базу знаний (использовать не менее 3 лингвистических переменных) для задачи выбора дозы снотворного (количество препарата, действие препарата, восприимчивость к выбранному препарату, цель и т.д.), проверить ее на полноту и произвести нечеткий вывод для конкретных значений (выбрать случайным образом).
15. Построить нечеткую базу знаний (использовать не менее 3 лингвистических переменных) для задачи планирования объема производства продукции (с учетом возможной прибыли, необходимых ресурсов, платежеспособности населения, рынка сбыта и т.д.), проверить ее на полноту и произвести нечеткий вывод для конкретных значений (выбрать случайным образом).
16. Построить нечеткую базу знаний (использовать не менее 3 лингвистических переменных) для задачи регулирования кондиционера (учитывать его мощность, объем помещения, температуру окружающей среды, необходимую температуру в помещении и т.д.), проверить ее на полноту и произвести нечеткий вывод для конкретных значений (выбрать случайным образом).
17. Построить нечеткую базу знаний (использовать не менее 3 лингвистических переменных) для задачи распределения нагрузки между компьютерами при использовании их в кластерах (учитывать характеристики компьютеров, их количество, количество параллельного кода, характеристики сети и т.д.), проверить ее на



полноту и произвести нечеткий вывод для конкретных значений (выбрать случайным образом).

18. Построить нечеткую базу знаний (использовать не менее 3 лингвистических переменных) для задачи выбора складского помещения (учитывать площадь склада, количество и размеры продукции, удаленность от места производства и точек реализации, свойства продукции и характеристики помещений и т.д.), проверить ее на полноту и произвести нечеткий вывод для конкретных значений (выбрать случайным образом).
19. Построить нечеткую базу знаний (использовать не менее 3 лингвистических переменных) для задачи выбора комплектующих для компьютера (учитывать цену, потребности пользователя, совместимость, сроки использования и т.д.), проверить ее на полноту и произвести нечеткий вывод для конкретных значений (выбрать случайным образом).
20. Построить нечеткую базу знаний (использовать не менее 3 лингвистических переменных) для задачи определения количества линий в службе поддержки (учитывать количество обслуживаемых клиентов, среднюю частоту обращения в службу одного клиента, среднее время обслуживания одной заявки, квалификацию персонала и т.д.), проверить ее на полноту и произвести нечеткий вывод для конкретных значений (выбрать случайным образом).

## ЛИТЕРАТУРА

1. Сидоркина И. Г. Системы искусственного интеллекта [Текст] : учебное пособие для вузов / И. Г. Сидоркина. – Москва : КНОРУС, 2011. – 246 с. – Гриф УМО. – ISBN 978-5-406-00449-4.
2. Ясницкий Л. Н. Искусственный интеллект. Элективный курс [Электронный ресурс]: учебное пособие / Л. Н. Ясницкий. – Москва: Бином. Лаборатория знаний, 2012. – 201 с. – ISBN 978-5-9963-1482-9. – Режим доступа:  
[http://e.lanbook.com/books/element.php?pl1\\_cid=25&pl1\\_id=8776](http://e.lanbook.com/books/element.php?pl1_cid=25&pl1_id=8776).
3. Ясницкий Л. Н. Введение в искусственный интеллект [Текст] : учебное пособие для студентов / Л. Н. Ясницкий. – 3-е изд., стер. – Москва : Издат. центр Академия, 2010. – 176 с. – ISBN 978-5-7695-7042-1.
4. Андрейчиков А. В. Интеллектуальные информационные системы [Текст] : учебник для вузов / А. В. Андрейчиков, О. Н. Андрейчикова. – М. : Финансы и статистика, 2006 . – 424 с – Гриф МО.
5. Глухих И. Н. Интеллектуальные информационные системы [Текст] : учебное пособие для вузов / И. Н. Глухих ; РФ МО и науки ГОУ ВПО Тюменский гос. ун-т. – Москва : Академия, 2010. – 112 с. – ISBN 978-5-7695-7089-6.
6. Рыбина Г. В. Основы построения интеллектуальных систем [Текст] : учебное пособие / Г.В. Рыбина. – М : Финансы и статистика, 2010. – 432 с. – ISBN 978-5-279-03412-3.
7. Липатова С. В. Сборник задач по курсу «Интеллектуальные информационные системы» [Текст]: учебное пособие / С.В. Липатова. – Ульяновск : Изд-во УлГУ, 2010. – 64 с.

---

Отпечатано в издательско-полиграфическом центре  
Набережночелнинского института  
Казанского (Приволжского) федерального университета

Подписано в печать 17.02.2017г.  
Формат 60x84/16. Печать ризографическая.  
Бумага офсетная. Гарнитура «Times New Roman»  
Усл.-печ.л. 2,9. Уч.-изд.л. 2,9.  
Тираж 50 экз. Заказ № 811.

---

423810, г. Набережные Челны, Новый город, проспект Мира, 68/19  
тел./факс (8552) 39-65-99 e-mail: [ic-nchi-kpfu@mail.ru](mailto:ic-nchi-kpfu@mail.ru)