МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ

(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)

Факультет информационных технологий и прикладной математики

Кафедра вычислительной математики и программирования

**«Логическое программирование»**

**Реферат**

**«Логические языки и базы данных»**

Студент: Иванов Д.С.

Группа: М8О-207Б-17

Руководитель: Сошников Д. В.

Оценка:

Дата:

Москва

2018г.

ВВЕДЕНИЕ. .………………………………………………………….….1

1. Основы логического программирования…………………………………….…….….…1
2. Язык программирования Prolog…..…………………………………………………….…..…2
3. Базы данных……………………………………………………..…3

ЗАКЛЮЧЕНИЕ…………………………………………………………...5

БИБЛИОГРАФИЧЕСКИЙ СПИСОК…………………………………………………………………..6

**ВВЕДЕНИЕ**

Информация становится со временем всё более актуальным ресурсом. Человек на протяжении всех времен старался придумать новые и улучшить уже имеющиеся приспособления. Но пиком этой гонки стало появление компьютера.

Компьютер, безусловно, стал главным помощником человека, но ведь чтобы его использовать, с ним нужно общаться. Для этого стали разрабатываться способы общения с компьютером — языки программирования. За все время появилось множество языков программирования, однако, из всей группы можно выделить два класса: императивные (процедурные) и декларативные (логические).

К императивным языкам относятся C, C++, Pascal, Java и другие. В основе этих языков лежит модель фон Неймана. Для решения задачи её необходимо смоделировать, и пошагово прописать решения (шаги). Такие языки, приближенные к машинному языку относят к языкам низкого уровня.

Ко второму классу языков программирования, декларативному, относятся такие языки, как Prolog, Mercury, Lisp. Они являются противоположностью императивным, так как при составлении модели мы описываем не способ получения решения, а описание задачи и само решение, каким мы хотим его видеть. То есть, предоставляем все вычисления компьютеру, и получаем язык более близкий к человеческому, чем к машинному, поэтому их можно назвать не просто языками высокого уровня, а даже выше.

**1.Основы логического программирования**

Логическое программирование — парадигма программирования, основанная на автоматическом доказательстве теорем, с использованием механизмов логического вывода информации на основе заданных фактов и правил вывода. Язык Пролог и логическое программирование широко используются для создания баз знаний и экспертных систем и исследований в сфере искусственного интеллекта на основе логических моделей баз знаний и логических процедур вывода и принятия решений.

Под Базами знаний в информатике понимают совокупность фактов и правил вывода, допускающих логический вывод и осмысленную обработку информации. В языке Пролог базы знаний описываются в форме конкретных фактов и правил логического вывода над базами данных и процедурами обработки информации, представляющих сведения и знания о людях, предметах, фактах событиях и процессах в логической форме.

Наиболее важным свойством информации, хранящейся в базах знаний, является достоверность конкретных и обобщенных сведений в базе данных и релевантности информации, получаемой с использованием правил вывода, заложенных в базу знаний.

В ответах на простейшие запросы к базам знаний системы логического программирования Пролог, выдает значения «истина» и «ложь» в зависимости от наличия соответствующих фактов. Истинность и ложность ответов существенно зависит от наполнения баз знаний фактической информацией и ее непротиворечивостью.

Обобщенные сведения в базах знаний на языке Пролог задаются с помощью правил логического вывода, выполняющих роль определения понятий, а также логических процедур, состоящих из наборов правил логического вывода. Достоверность обобщенных сведений зависит от наличия необходимых фактов и достоверности данных в базах знаний.

**2. Prolog как язык программирования**

Чтобы лучше понять основы логических языков программирования попробуем рассмотреть самого яркого их представителя — Prolog.

Язык сосредоточен вокруг небольшого набора основных механизмов, включая сопоставление с образцом, древовидного представления структур данных и автоматического перебора с возвратами. Хорошо подходит для решения задач, где рассматриваются объекты и отношения между ними. Пролог, благодаря своим особенностям, используется в области искусственного интеллекта, компьютерной лингвистики и нечислового программирования в целом. В некоторых случаях реализация символьных вычислений на других стандартных языках вызывает необходимость создавать большое количество кода, сложного в понимании, в то время как реализация тех же алгоритмов на языке Пролог даёт простую программу, легко помещающуюся на одной странице.

Prolog является декларативным языком программирования: логика программы выражается в терминах отношений, представленных в виде фактов и правил. Для того чтобы инициировать вычисления, выполняется специальный запрос к базе знаний, на которые система логического программирования генерирует ответы «истина» и «ложь». Для обобщённых запросов с переменными в качестве аргументов созданная система Пролог выводит конкретные данные в подтверждение истинности обобщённых сведений и правил вывода.

**3. Базы данных**

В основе любой базы данных лежит модель данных. Модель данных — совокупность структур данных и их отношений. Для систем управления баз данных (СУБД) используются три основные модели: иерархическая, сетевая и реляционная, а также их комбинации. Определим каждый тип модели данных.

Иерархическая. Представляет собой простое дерево, в котором есть уровни, узлы и связи. В вершине дерева находится единственный узел — корень. Основное правило, что у каждой вершины только один предок, а количество потомков от нуля до нескольких. Между узлами установлена целая связь, между различными деревьями связь не поддерживается. Для обработки иерархических баз данных используют методы в глубину и в ширину.

Сетевая. Они представляют собой более сложный вариант иерархических моделей. В ней все те же элементы, однако, есть отличие в связях. В сетевых моделях у узла может быть более одного предка.

Реляционная. Из-за неполноты иерархической и сетевой моделей появилась реляционная модель данных. В основе реляционного подхода то, что все данные хранятся в двумерных таблицах (или же переводя на язык программирования — двумерный массив) и пользователь работает с операторами для создания баз данных. Таким образом, «Реляционные базы данных — единое хранилище данных, которое однозначно определяется, а затем используется многими пользователями. Изменение и добавление данных в базу данных не влияет на приложение.»[14]. Реляционные базы данных лежали почти во всех продуктах с конца 70-х годов, однако сейчас активно развивается новый тип баз данных: дедуктивный.

Дадим определение дедуктивной базе данных, «дедуктивная база данных — это система баз данных, которая может делать выводы (то есть заключать дополнительные факты) на основе фактов и правил хранящихся в (дедуктивной базе данных). Их цель объединить логическое программирование и реляционные базы данных (базы данных, в которых есть отношение) с целью построения систем, способных обрабатывать большие объемы данных. Для этого в качестве основного языка выступает язык — Datalog» [13].

Так почему же Prolog не стал основным языком для развития дедуктивных баз данных, а создали альтернативный Prolog язык Datalog, ведь как оказалось Prolog очень функционален?

В плане создания дедуктивных баз данных Prolog имеет такие минусу как «покортежная (кортеж – упорядоченный набор фиксированной длины) обработка (Prolog возвращает отдельные кортежи, а не множество кортежей); процедурность и чувствительность к порядку записи правил или фактов; потеря декларативности языка из-за специальных предикатов»[12]. Datalog в противовес Prolog наоборот является полным языком баз данных: непроцедурным, множественным, без специальных предикатов и не важен порядок записи фактов и правил. Также стоит отметить, что в Prolog интерпретация нисходящая (поиск сначала вглубь, а потом бектрекинг с конструированием деревьев доказательств), а в Datalog интерпретация восходящая (вычисление неподвижной точки программы).

В настоящее разрабатывается множество исследовательских систем, в основе которых лежит связь реляционных баз данных и логическое программирование, например LDL (создание системы для обработки запросов на логическом языке), NAIL! (обработка запросов на Datalog, но с традиционными СУБД), PRISMA (поддержание логического интерфейса), ALGRES (поддержка ненормализованных отношений и операций вычисления неподвижной точки), ADE (разработка баз данных на основе объектно-ориентированном языке), ICOT (выбор Prolog как основы для разработки архитектуры вычислительных систем, стиля программирования и баз данных), POSTGRES (разработка баз данных нового поколения) [12].

**ЗАКЛЮЧЕНИЕ**

Таким образом, мы рассмотрели взаимосвязь логического программирования и баз данных. И хотя мы эти направления существуют совсем недолго, однако они стремительно развиваются. Примером этому служат начальные области применения искусственного интеллекта в наших сферах жизни: финансах (способны проводить быстро и эффективно огромное количество вычислений), промышленности (эффективны в рутинной работе, где необходима точность), онлайн-применение (боты и помощники), распределение ресурсов и множество других. Однако это ещё не предел и улучшения все ещё ведутся, как было приведено выше улучшаются не только базы данных, но и языки логического программирования, ведь развитие ИИ сегодня одно из самых перспективных направлений.

**Библиографический список:**

1. <http://calcsbox.com/post/matematiceskaa-logika-i-iskusstvennyj-intellekt.html> — математическая логика и искусственный интеллект
2. <https://habr.com/post/324688/> — отличие императивных и декларативных языков
3. <https://helpiks.org/4-63413.html> — классификация языков программирования
4. <https://timeweb.com/ru/community/articles/bazy-dannyh-i-subd-1> — базы данных и СУБД
5. <https://studopedia.ru/3_82186_logicheskoe-programmirovanie-i-bazi-dannih.html> — логическое программирование и базы данных
6. <http://labs.org.ru/prolog-1/> — язык логического программирования Prolog
7. <http://rusinov.net/wp-content/uploads/2012/05/%D0%9E%D1%81%D0%BD%D0%BE%D0%B2%D1%8B-%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F-%D0%BD%D0%B0-%D0%9F%D1%80%D0%BE%D0%BB%D0%BE%D0%B3.pdf> — основы программирования на языке Prolog
8. <http://bourabai.ru/alg/prolog.htm> — логическое программирование на языке Prolog
9. <https://habr.com/post/124636/> — Prolog удивительный язык программирования
10. <https://pro-prof.com/archives/2362> — введение в логическое программирование(Prolog)
11. <http://fb.ru/article/412294/logicheskie-yazyiki-programmirovaniya-osobennosti-primeryi> — что такое логические языки программирования?
12. С. Чери, Г. Готлоб, Л. Танка — «Логическое программирование и базы данных».
13. <https://ru.wikipedia.org/wiki/%D0%94%D0%B5%D0%B4%D1%83%D0%BA%D1%82%D0%B8%D0%B2%D0%BD%D0%B0%D1%8F_%D0%B1%D0%B0%D0%B7%D0%B0_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85> — дедуктивные базы данных
14. <https://studfiles.net/preview/5389850/page:3/> — модели данных
15. <https://ru.wikipedia.org/wiki/%D0%98%D1%81%D0%BA%D1%83%D1%81%D1%81%D1%82%D0%B2%D0%B5%D0%BD%D0%BD%D1%8B%D0%B9_%D0%B8%D0%BD%D1%82%D0%B5%D0%BB%D0%BB%D0%B5%D0%BA%D1%82> — искусственный интелект