**Журавлёв Д. ИСБ-119**

**Ответы на КВ лекция 02.12.2021**

**1. Привести описание «языков программирования» ЭС, стандартов и диалектов.**

Согласно ГОСТ 19.506-79 структура описания языка программирования должна быть следующей:

1. Общие сведения.

Пролог (англ. Prolog) — язык и система логического программирования, основанные на языке предикатов математической логики дизъюнктов Хорна, представляющей собой подмножество логики предикатов первого порядка.

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.

2. Элементы языка.

Терминология. Объекты данных в Прологе называются термами.

Термы бывают следующих видов:

* Константами. Делятся на числа и атомы. Начинаются с маленькой буквы.
* Числа: 1,36, 0, -1, 123.4, 0.23E-5.
* Атомы — это просто символы и строки: a, abc, neOdinSimvol, sTROKa. Если атом состоит из пробела, запятых и тд, то нужно их обрамлять в одинарные кавычки. Пример атома: 'строка с пробелами, запятыми. Eto kirilicca'.
* Переменными. Начинаются с заглавной буквы: X, Y, Z, Peremennaya, Var.
* Структурами (сложные термы). Например, study(misha, lp).
* Списками. Пример: [X1], [Head|Tail]. Списки — важная структура в Прологе. Списки позволяют хранить произвольное количество данных. Связный список — структура данных, состоящая из узлов. Узел содержит данные и ссылку (указатель, связку) на один или два соседних узла. Списки языка Prolog являются односвязными, т.е. каждый узел содержит лишь одну ссылку.

УТВЕРЖДЕНИЯ

Программа на Прологе есть совокупность утверждений. Утверждения состоят из целей и хранятся в базе данных Пролога. Таким образом, база данных Пролога может рассматриваться как программа на Прологе. В конце утверждения ставится точка “.”. Иногда утверждение называется предложением.

Основная операция Пролога - доказательство целей, входящих в утверждение.

Существуют два типа утверждений:

* факт: это одиночная цель, которая, безусловно, истинна;
* правило: состоит из одной головной цели и одной или более хвостовых целей, которые истинны при некоторых условиях.

Правило обычно имеет несколько хвостовых целей в форме конъюнкции целей.

Конъюнкцию можно рассматривать как логическую функцию И. Таким образом, правило согласовано, если согласованы все его хвостовые цели.

Примеры фактов:

* собака(реке). родитель(голди.рекс).

Примеры правил:

* собака (X) :- родитель (X.Y),собака (Y). человек(Х) :-мужчина(Х).

ЗАПРОСЫ

После записи утверждений в базу данных вычисления могут быть инициированы вводом запроса.

Запрос выглядит так же, как и целевое утверждение, образуется и обрабатывается по тем же правилам, но он не входит в базу данных (программу). В Прологе вычислительная часть программы и данные имеют одинаковый синтаксис. Программа обладает как декларативной, так и процедурной семантикой.

Запрос обозначается в Прологе утверждением ?-, имеющим арность 1. Обычно запрос записывается в операторной форме: за знаком ?- следует ряд хвостовых целевых утверждений (чаще всего в виде конъюнкции).

Приведем примеры запроса:

* ?-собака(X). ?- родитель(Х.У),собака (Y). или, иначе,
* '?-'(собака(Х)) С?-') ','(родитель(Х„У”,собака (Y)).

3. Методы структурирования программы.

Введение списка утверждений в Пролог-систему осуществляется с помощью встроенного предиката consult. Аргументом предиката consult является атом, который обычно интерпретируется системой как имя файла, содержащего текст программы на Прологе. Файл открывается, и его содержимое записывается в базу данных. Если в файле встречаются управляющие команды, они сразу же выполняются. Возможен случай, когда файл не содержит ничего, кроме управляющих команд для загрузки других файлов. Для ввода утверждений с терминала в большинстве реализации Пролога имеется специальный атом, обычно user. С его помощью утверждения записываются в базу данных, а управляющие команды выполняются немедленно.

Помимо предиката consult, в Прологе существует предикат reconsult. Он работает аналогичным образом. Но перед добавлением утверждений к базе данных из нее автоматически удаляются те утверждения, головные цели которых сопоставимы с целями, содержащимися в файле перезагрузки. Такой механизм позволяет вводить изменения в базу данных. В Прологе имеются и другие методы добавления и удаления утверждений из базы данных. Некоторые реализации языка поддерживают модульную структуру, позволяющую разрабатывать модульные программы.

В заключение раздела дадим формальное определение синтаксиса Пролога, используя форму записи Бэкуса-Наура/

запрос ::- голова утверждения

правило ::– голова утверждения :- хвост утверждения

факт ::- голова утверждения

голова утверждения ::-атом | структура

хвост утверждения ::- атом структура,

термы ::-терм [,термы]

терм ::- число | переменная | атом | структура

структура ::-атом (термы)

**2. Какие задачи могут решаться при использовании выбранных языков.**

Пролог удобен в решении задач, в которых мы знаем начальное состояние (объекты и отношения между ними) и в которых нам трудно задать четкий алгоритм поиска решений. То есть чтобы Пролог сам нашел ответ.

Список задач, в которых Пролог удобен:

* Искусственный интеллект
* Компьютерная лингвистика. Написание стихов, анализ речи
* Поиск пути в графе. Работа с графами
* Логические задачи
* Нечисловое программирование

3. Схожесть и отличие от CLIPS.

Понятия «цель» в CLIPS не существует. Схема, показанная на рисунке 1 иллюстрирует общую архитектуру системы.

Если ПРОЛОГ не может доказать цель, он предполагает, что цель ложна — это называется предположением о закрытом мире (closed-world assumption). В любом случае задача достижения цели остается на усмотрение механизма вывода.

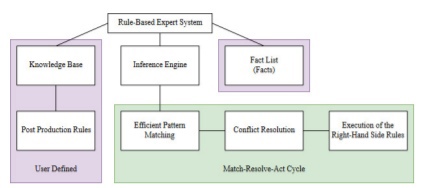


Рисунок 1 – архитектура системы