|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ «Информатика и системы управления»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»\_\_\_\_\_\_\_\_\_\_\_\_

*Лабораторная работа №11*

*По предмету: «Функциональное и логическое программирование»*

Преподаватели: Толпинская Н. Б.,

Строганов Ю. В.,

Студент: Мирзоян С.А.,

Группа: ИУ7-65Б

Москва, 2020 г.

Теоретическая часть

Prolog — это декларативный язык программирования, при использовании которого решение задачи получается путем логического вывода из ранее известных положений. Программа на Prolog не является определением последовательности действий, она представляет собой набор фактов и правил, обеспечивающих получение заключений на основе этих утверждений. Prolog базируется на предложениях Хорна, являющихся подмножеством формальной системы, которая называется логикой предикатов.

Программа на Prolog представляет собой: *базу знаний*и *вопрос*. С помощью подбора ответов на запросы он (Prolog, программа) извлекает хранящуюся (известную в программе) информацию. База знаний содержит истинностные знания, используя которые программа выдает ответ на запрос. Одной из особенностей Prolog является то, что при поиске ответов на вопрос, он рассматривает альтернативные варианты и находит все возможные решения (методом проб и ошибок) — множества значений переменных, при которых на поставленный вопрос можно ответить – «да».

База знаний состоит из предложений - CLAUSES (отдельных знаний или утверждений): фактов и правил. Каждое предложение заканчивается точкой. Предложения бывают двух видов: факты и правила. Предложение более общего вида – правило имеет вид:

A :- B1,... , Bn.

A называется заголовком правила, а B1,..., Bn – телом правила.

Вопрос состоит только из тела – составного терма (или нескольких составных термов). Вопросы используются для выяснения выполнимости некоторого отношения между описанными в программе объектами. Система рассматривает вопрос как цель, к которой (к истинности которой) надо стремиться. Ответ на вопрос может оказаться логически положительным или отрицательным, в зависимости от того, может ли быть достигнута соответствующая цель.

Поиск содержательного ответа на поставленный вопрос, с помощью имеющейся базы знаний, фактически заключается в поиске нужного знания, но какое знание понадобится – заранее неизвестно. Этот поиск осуществляется формально с помощью механизма унификации, встроенного в систему и не доступного программисту. Упрощенно, процесс унификации можно представить как формальный процесс сравнивания (сопоставления) терма вопроса с очередным термом знания. При этом, знания по умолчанию просматриваются сверху вниз, хотя такой порядок и не очевиден. В процессе сравнивания для переменных «подбираются», исходя из базы знаний, значения (для именованных переменных). И эти подобранные для переменных значения возвращаются в качестве побочного эффекта ответа на поставленный вопрос. В вопросы переменные входят только с квантором существования, а в факты и правила входят только с квантором всеобщности.

**Цель работы программы:** Определить является ли вопрос логическим следствием программы или нет, что выполняется с применения правил вывода. Правила вывода – это утверждения о взаимосвязи между допущениями и заключениями, которые с позиции исчисления предикатов верны всегда.

Для выполнения логического вывода используется механизм (алгоритм) унификации, встроенный в систему.

Унификация – операция, которая позволяет формализовать процесс логического вывода (наряду с правилом резолюции). С практической точки зрения - это основной вычислительный шаг, с помощью которого происходит:

1. Двунаправленная передача параметров процедурам,
2. Неразрушающее присваивание,
3. Проверка условий (доказательство).

В процессе работы система выполняет большое число унификаций. Процесс унификации запускается автоматически, но пользователь имеет право запустить его принудительно с помощью утверждения (немного нарушает форму записей): T1 = T2 . Унификация – попытка "увидеть одинаковость" – сопоставимость двух термов, может завершаться успехом или тупиковой ситуацией (неудачей). В последнем случае включается механизм отката к предыдущему шагу.

Практическая часть

Тестовая программа.

Листинг.

1. predicates
2. likes(symbol,symbol)
3. clauses
4. likes(ellen,tennis).
5. likes(john,football).
6. likes(tom,baseball).
7. likes(eric,swimming).
8. likes(mark,tennis) .
9. likes(bill,Activity):-likes (tom, Activity) .
10. goal
11. likes(bill, baseball).

Ответ “Yes”

Следующий запрос в GOAL-разделе: likes(bill, football) приводит к ответу “No”

Пролог ищет ответ на запрос проходя по фактам сверху вниз. Просматривается каждый терм знания, пока не будет достигнут конец списка. В данной программе присутствует правило, которое можно трактовать как: «Билл любит все те активности, которые любит Том». Так как Том любит бейсбол, то и Билл его тоже любит, программа вывела «yes». Так как Том не любит футбол, то и Билл его не любит, и возвращаемый ответ будет «no».

Программа «Телефонный справочник».

Листинг:

1. predicates
2. client(symbol, symbol, symbol).
3. clauses
4. client("Mary","89259314541","Megafon").
5. client("Ivan","87773451234","Beeline").
6. client("Oleg","89059933257","Tele2").
7. client("Nikolay","89423934454","MTS").
8. client("Igor","89231256739","Yota").
9. client("Ira","89863123521","Megafon").
10. client("Luba","8324876637","Beeline").
11. goal
12. client(name, number, **operator**).