**Цели данных лабораторных работ**: обучение практическим навыкам программирования на языке Prolog — одном из самых распространенных языков 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 Prolog и логическое программирование широко используются для создания баз знаний, экспертных систем и исследований в сфере искусственного интеллекта на основе логических моделей баз знаний и логических процедур вывода и принятия решений.

**Поставленные задачи**: освоение методов и принципов программирования на языке Prolog и получения навыков работы со средой Turbo Prolog.

[Из методички]

Теоретическая часть

1. Что такое язык программирования? Какие бывают? В чем их отличия
2. Подробнее про логическое программирование. История. Основы работы
3. Подробнее про Пролог. Логическое программирование=Пролог? Или есть другие языки логического программирования?

Язык программирования – формальный знаковый язык, с помощью которого создаются компьютерные программы, он определяет внешний вид программы и то, что машина будет делать. (микс [https://ru.wikipedia.org](https://ru.wikipedia.org/wiki/%D0%AF%D0%B7%D1%8B%D0%BA_%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%9A%D0%BB%D0%B0%D1%81%D1%81%D0%B8%D1%84%D0%B8%D0%BA%D0%B0%D1%86%D0%B8%D1%8F) и <http://progopedia.ru/>)

На сегодняшний день существует тысячи языков программирования. Абсолютно разные языки, выполняют разные задачи. Существует разделение на языки высокого и низкого уровня.

К языкам программирования высокого уровня относятся языки, которые больше понятны программисту, чем машине. Здесь не учитываются особенности архитектур компьютера. С помощью высокоуровневых языков программирования можно решить задачу из любой области. Программистам нет необходимости знать, на какой ЭВМ будет запущена программа. Например, Python, C++, JavaScript, Ruby и т.д.

Низкоуровневые языки программирования разрабатываются для конкретной архитектуры компьютера. Программы, написанные на низкоуровневом языке, выполняются быстрее тех, что написаны на высокоуровневом, так как такие программы не нужно интерпретировать или компилировать, они взаимодействуют непосредственно с памятью компьютера и его регистрами. Примеры языков – Ассемблер, С.

Помимо разделения на уровни, языки программирования делятся на процедурные и декларативные.

Процедурное программирование представляется собой данные, которые хранятся в переменных, и последовательность команд, которая эти данные обрабатывает. Эти команды явно и однозначно задают преобразования, которые необходимо выполнить над данными.

Базой декларативного программирования является булева алгебра и формальная логика. В программе, написанной на декларативной языке, нет явных алгоритмов. Есть условия и отношения, по которым программа строит дерево вывод и в последствии находи решение.

**Декларативные языки программирования** - это языки объявлений и построения структур. К ним относятся функциональные и логические языки программирования. В этих языках не производится алгоритмических действий явно, то есть алгоритм не задается прграммистом, а строится самой программой. В декларативных языках задается, производится построение какой-либо структуры или системы, то есть декларируются (объявляются) какие-то свойства создаваемого объекта. Эти языки получили широкое применение в системах автоматизированного проектирования (САПР), в так называемых CAD-пакетах, в моделировнии, системах исккусственного интеллекта.

Непроцедурное (декларативное) программирование появилось в начале 70-х годов 20 века, К непроцедурному программированию относятся функциональные и логические языки.

В функциональных языках программа описывает вычисление некоторой функции. Обычно эта функция задается как композиция других, более простых, те в свою очередь делятся на еще более простые задачи и т.д. Один из основных элементов функциональных языков – рекурсия. Оператора присваивания и циклов в классических функциональных языках нет.

ПРОЛОГ — язык ДЕКЛАРАТИВНЫЙ. Он базируется на естественных для человека логических принципах. Нужно уметь составить формальное описание задачи, используя понятия объектов различных типов и отношений между ними. Иными словами, нужно описать все ФАКТЫ (ИСТИННЫЕ УТВЕРЖДЕНИЯ) и ПРАВИЛА (позволяющие ВЫВЕСТИ из уже имеющихся истинных утверждений новые), описывающие данную ситуацию. Затем пользователь задает вопрос или, пользуясь терминологией Пролога, задает ЦЕЛЬ.

Далее Пролог пытается ДОКАЗАТЬ заданную цель, то есть вывести ее логическим путем из уже имеющихся фактов и правил. Этот логический вывод осуществляют ВСТРОЕННЫЕ УНИФИКАЦИОННЫЕ ПРОЦЕДУРЫ путем перебора имеющихся утверждений и попыток сопоставить их с доказываемой целью. В Прологе отсутствуют явные управляющие структуры типа IF...THEN...ELSE..., нет детального описания последовательности шагов. Все это — дело внутренних процедур. Иногда Пролог считают не языком программирования, а пользовательской оболочкой, позволяющей формулировать запросы к базе данных, записанной в Пролог — программе.

**Логические языки программирования**

Языки программирования данного типа основываются на формальной логике и булевой алгебре. Программа не содержит в себе

**Функциональные языки программирования**

Функциональное программирование основывается на использование списков и функций. Переменные могут отсутствовать вообще.

Prolog (фр. Programmation en Logique) — это язык логического программирования, основанный на языке предикатов математической логики дизъюнктов Хорна — подмножестве логики предикатов первого порядка. Существует несколько различных сред и компиляторов для программирования на языке Prolog, например:

* Turbo Prolog — язык и система логического программирования, разработанные компанией Borland в начале 1980-х гг.;
* Visual Prolog — объектно-ориентированное расширение языка программирования PDC Prolog, развивавшегося из Turbo Prolog, а также система визуального программирования;
* SWI-Prolog — открытая реализация языка Prolog.

В данных лабораторных работах будет описываться работа в среде Turbo Prolog.

Ссылки

<http://progopedia.ru/language/prolog/>

<http://www.learnprolognow.org/>

<https://www.metalevel.at/prolog>

<https://drive.uqu.edu.sa/_/fbshareef/files/textbook.pdf>

<http://tka4.org/materials/lib/Articles-Books/Programming/Perl,Prolog,Python,/%D0%9A%D0%BB%D0%BE%D0%BA%D1%81%D0%B8%D0%BD,%D0%9C%D0%B5%D0%BB%D0%BB%D0%B8%D1%88,_%D0%9F%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%D0%B5_%D0%BD%D0%B0_%D1%8F%D0%B7%D1%8B%D0%BA%D0%B5_%D0%9F%D0%A0%D0%9E%D0%9B%D0%9E%D0%93.pdf>

<http://progbook.ru/prolog>