# Разработка синтаксиса проблемно-ориентированного языка программирования

Разработка синтаксиса проблемно-ориентированного языка. Лабораторная работа выполняется в группах по два человека. Целью работы является ознакомление с технологиями разработки синтаксиса языка программирования. Последовательность действий: анализ предметной области, формализация предметной области в виде концептов языка программирования (абстрактный синтаксис), разработка синтаксического представления концептов предметной области (конкретный синтаксис).

## Задания

### 1. Разработка синтаксиса для подмножества языка Паскаль

1. Проектирование синтаксиса
   1. Проанализировать предметную область подмножества языка программирования Паскаль.
   2. Выделить множество конструкций языка, подлежащих реализации: программа, переменная, константа, встроенные и пользовательские типы, функция, процедура, условный оператор, операторы циклов, арифметические и логические операторы и т.д.
   3. Для выделенного множества конструкций разработать иерархию концептов, определив их свойства и отношения ассоциации и агрегации между ними.
2. Реализация синтаксиса
   1. Создать проект «Pascal.mpr» в среде метапрограммирования MPS.
   2. В проекте «Pascal.mpr» создать язык «ru.ifmo.pascal».
   3. В модели «structure» языка «ru.ifmo.pascal» завести разработанные концепты со свойствами и отношениями.
   4. В модели «editor» языка «ru.ifmo.pascal» определить для концептов редакторы соответствующие конкретному синтаксису языка Паскаль.
3. Тестирование синтаксиса
   1. В проекте «Pascal.mpr» создать решение (solution) «ru.ifmo.pascal.sandbox».
   2. В решении «ru.ifmo.pascal.sandbox» завести модель «ru.ifmo.pascal.sample».
   3. В модели «ru.ifmo.pascal.sample» привести примеры использования всех конструкций разработанного языка «ru.ifmo.pascal».
   4. Убедиться в том, что все конструкции корректно вводятся и отображаются в редакторе среды MPS.

### 2. Разработка синтаксиса для языка Лисп

1. Проектирование синтаксиса
   1. Проанализировать предметную область языка программирования Лисп.
   2. Выделить множество конструкций языка, подлежащих реализации: символьное выражение, определение и вызов функции, список, атом, пустой список, CONS, CAR, CDR, точечная нотация и т.д.
   3. Для выделенного множества конструкций разработать иерархию концептов, определив их свойства и отношения ассоциации и агрегации между ними.
2. Реализация синтаксиса
   1. Создать проект «Lisp.mpr» в среде метапрограммирования MPS.
   2. В проекте «Lisp.mpr» создать язык «ru.ifmo.lisp».
   3. В модели «structure» языка «ru.ifmo.lisp» завести разработанные концепты со свойствами и отношениями.
   4. В модели «editor» языка «ru.ifmo.lisp» определить для концептов редакторы соответствующие конкретному синтаксису языка Лисп.
3. Тестирование синтаксиса
   1. В проекте «Lisp.mpr» создать решение (solution) «ru.ifmo.lisp.sandbox».
   2. В решении «ru.ifmo.lisp.sandbox» завести модель «ru.ifmo.lisp.sample».
   3. В модели «ru.ifmo.lisp.sample» привести примеры использования всех конструкций разработанного языка «ru.ifmo.lisp».
   4. Убедиться в том, что все конструкции корректно вводятся и отображаются в редакторе среды MPS.

### 3. Разработка синтаксиса для языка Пролог

1. Проектирование синтаксиса
   1. Проанализировать предметную область языка программирования Пролог.
   2. Выделить множество конструкций языка, подлежащих реализации: имя, переменная, список, терм, атом, факт и т.д.
   3. Для выделенного множества конструкций разработать иерархию концептов, определив их свойства и отношения ассоциации и агрегации между ними.
2. Реализация синтаксиса
   1. Создать проект «Prolog.mpr» в среде метапрограммирования MPS.
   2. В проекте «Prolog.mpr» создать язык «ru.ifmo.prolog».
   3. В модели «structure» языка «ru.ifmo.prolog» завести разработанные концепты со свойствами и отношениями.
   4. В модели «editor» языка «ru.ifmo.prolog» определить для концептов редакторы соответствующие конкретному синтаксису языка Пролог.
3. Тестирование синтаксиса
   1. В проекте «Prolog.mpr» создать решение (solution) «ru.ifmo.prolog.sandbox».
   2. В решении «ru.ifmo.prolog.sandbox» завести модель «ru.ifmo.prolog.sample».
   3. В модели «ru.ifmo.prolog.sample» привести примеры использования всех конструкций разработанного языка «ru.ifmo.prolog».
   4. Убедиться в том, что все конструкции корректно вводятся и отображаются в редакторе среды MPS.

### 4. Разработка синтаксиса для языка разметки HTML

1. Проектирование синтаксиса
   1. Проанализировать предметную область языка разметки HTML.
   2. Выделить множество конструкций языка, подлежащих реализации: документ, заголовок, тело, связь, ссылка, блоки, атрибуты и т.д.
   3. Для выделенного множества конструкций разработать иерархию концептов, определив их свойства и отношения ассоциации и агрегации между ними.
2. Реализация синтаксиса
   1. Создать проект «HTML.mpr» в среде метапрограммирования MPS.
   2. В проекте «HTML.mpr» создать язык «ru.ifmo.html».
   3. В модели «structure» языка «ru.ifmo.html» завести разработанные концепты со свойствами и отношениями.
   4. В модели «editor» языка «ru.ifmo.html» определить для концептов редакторы соответствующие конкретному синтаксису языка HTML.
3. Тестирование синтаксиса
   1. В проекте «HTML.mpr» создать решение (solution) «ru.ifmo.html.sandbox».
   2. В решении «ru.ifmo.html.sandbox» завести модель «ru.ifmo.html.sample».
   3. В модели «ru.ifmo.html.sample» привести примеры использования всех конструкций разработанного языка «ru.ifmo.html».
   4. Убедиться в том, что все конструкции корректно вводятся и отображаются в редакторе среды MPS.

### 5. Разработка синтаксиса для языка запросов SQL

1. Проектирование синтаксиса
   1. Проанализировать предметную область языка запросов SQL.
   2. Выделить множество конструкций языка, подлежащих реализации:
      1. Элементы DDL: операторы создания, модификации и удаления таблиц, атрибутов, доменов.
      2. Элементы DML: операторы выборки, удаления, модификации данных.
   3. Для выделенного множества конструкций разработать иерархию концептов, определив их свойства и отношения ассоциации и агрегации между ними.
2. Реализация синтаксиса
   1. Создать проект «SQL.mpr» в среде метапрограммирования MPS.
   2. В проекте «SQL.mpr» создать язык «ru.ifmo.sql».
   3. В модели «structure» языка «ru.ifmo.sql» завести разработанные концепты со свойствами и отношениями.
   4. В модели «editor» языка «ru.ifmo.sql» определить для концептов редакторы соответствующие конкретному синтаксису языка SQL.
3. Тестирование синтаксиса
   1. В проекте «SQL.mpr» создать решение (solution) «ru.ifmo.sql.sandbox».
   2. В решении «ru.ifmo.sql.sandbox» завести модель «ru.ifmo.sql.sample».
   3. В модели «ru.ifmo.sql.sample» привести примеры использования всех конструкций разработанного языка «ru.ifmo.sql».
   4. Убедиться в том, что все конструкции корректно вводятся и отображаются в редакторе среды MPS.