

ОЛИМПИАДА ШКОЛЬНИКОВ «ШАГ В БУДУЩЕЕ»

НАУЧНО-ОБРАЗОВАТЕЛЬНОЕ СОРЕВНОВАНИЕ «ШАГ В БУДУЩЕЕ, МОСКВА»

Техника и технологии - программирование

название факультета

Программное обеспечение ЭВМ и информационные технологии

название кафедры

Создание собственного интерпретатора одного из диалектов языка
программирования Lisp для написания специализированных скриптов

название работы

Автор:

Диникеев Аскар Маратович

фамилия, имя, отчество

МБОУ Лицей №83 г. Уфа 10 кл

наименование учебного заведения, класс

Научный руководитель:

Гильдин Александр Григорьевич

фамилия, имя, отчество

МБОУ Лицей №83 г. Уфа

место работы

Преподаватель

звание, должность

Москва - 2020

СОДЕРЖАНИЕ

АННОТАЦИЯ	3
ВВЕДЕНИЕ.....	4
1 Интерпретатор	5
1.1 Выбор Lisp в качестве языка программирования	5
1.2 Типы данных	5
1.3 Процесс интерпретации.....	6
2 Android приложение.....	7
2.1 Результат выполнения выражения	7
3 Примеры скриптов для приложения	7
ЗАКЛЮЧЕНИЕ	9
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ.....	10
Приложение А. Иерархия типов данных языка	11
Приложение Б. Процесс интерпретации скрипта интерпретатором.....	11
Приложение В. Пример создания и использования макроса в языке	11
Приложение Г. Пример раскрытия специальных форм	12
Приложение Д. Главное меню приложения	12
Приложение Е. Новостная лента при выполнении скрипта	13
Приложение Ж. Различные способы отображения результата выполнения.....	14

АННОТАЦИЯ

Существуют такие новостные форматы, как RSS и ATOM. При помощи них пользователь может организовать все свои новостные источники в одном приложении - агрегаторе. Некоторые сайты не поддерживают эти новостные форматы. Для решения данной проблемы был “с нуля” создан интерпретатор к собственному диалекту языка программирования Lisp, а также Android приложение для взаимодействия с интерпретатором.

В качестве языка программирования был выбран Lisp по следующим причинам: простота синтаксиса и мощная система макросов, достигающаяся за счет того, что код и данные в языке взаимозаменяемы.

Созданный язык и интерпретатор поддерживает такие продвинутые возможности, как закрытия, квазицитирование, макросы и HTTP запросы, позволяющие обрабатывать динамические вебсайты, получающие данные через JavaScript.

Созданное приложение позволяет пользователю редактировать, выполнять и просматривать результат выполнения скрипта в двух форматах: формат списка постов и текстовый формат.

В ходе проведенной работы была решена задача с чтением новостей с сайтов не поддерживающих протокол RSS, а также был создан свой собственный интерпретатор, позволяющий проводить дальнейшие исследования интерпретируемых языков программирования.

ВВЕДЕНИЕ

Существуют такие новостные форматы как RSS и Atom. Главное их преимущество для читателя заключается в отсутствии необходимости заходить на каждый отдельный сайт отслеживать новые публикации. Некоторые сайты не поддерживают эти новостные форматы.

Возникает противоречие между желанием использовать произвольные сайты в формате новостных лент и отсутствии на некоторых из них такой возможности. С целью разрешения данного противоречия была сформулированы цель и задачи работы.

Цель работы - создание приложения позволяющего читать новости с любого сайта в формате новостной ленты, в том числе с сайтов, не поддерживающих новостные протоколы. Указанный функционал должен быть реализован за счет написания пользовательских скриптов для собственного интерпретатора.

Задачи:

- Спроектировать и реализовать свой диалект языка Lisp и интерпретатор к нему “с нуля”. Не используя сторонние библиотеки и реализации.
- Создать Android приложение для взаимодействия с интерпретатором.

В ходе проделанной работы был создан интерпретатор для собственного диалекта языка программирования Lisp, а также приложение для создания и выполнения скриптов. Данные скрипты могут быть использованы для получения новостной ленты желаемого вебсайта. Помимо этого скрипт может быть использован для решения любой задачи.

1 Интерпретатор

1.1 Выбор Lisp в качестве языка программирования

Причина выбора Lisp в качестве языка программирования:

- Lisp и его диалекты очень просты в плане синтаксиса.
 - Это свойство позволяет мне больше сконцентрироваться на написании интерпретатора, а не на написании парсера.
- Мощная система макросов.
 - Макросы позволяют пользователю увеличивать возможности языка, не меняя исходного кода интерпретатора.

В результате проведенной работы был самостоятельно создан свой диалект языка программирования Lisp на языке программирования Kotlin, без использования сторонних библиотек и реализаций.

Созданный язык и интерпретатор поддерживают:

- Функции первого класса
- Закрывания
- Цитирование и квазичитирование выражений
- Макросы

Синтаксис созданного языка схож с синтаксисом Lisp языка Scheme. Главным отличием от Scheme является использование обычных списков, вместо пар объектов.

1.2 Типы данных

Язык содержит 7 типов данных:

- Логические значения (Boolean)
- Числа
- Строки: последовательность символов, заключенная в кавычки; поддерживает “escape characters” (\n, \r, \t, \", etc).
- Символы: имя переменной
- Nil

- Процедуры
 - Примитивные процедуры: процедуры, описанные в коде интерпретатора.
 - Композитные (пользовательские) процедуры: процедуры описанные пользователем.
- Списки

1.3 Процесс интерпретации

Процесс интерпретации скрипта можно разделить на 4 стадии:

- Токенизация: токенизатор выделяет из скрипта ключевые объекты в виде строк при помощи регулярных выражений.
- Парсинг: парсер превращает токены и последовательности токенов в выражения (Expressions).
- Раскрытие (Expansion): интерпретатор раскрывает значения некоторых синтаксических форм, а также объявляет и выполняет макросы.
- Выполнение (Evaluation): собственно выполнение выражений интерпретатором.


```

(defmacro (find-xml-tag str tag)
  (define tag1 (string-append "<" tag ">"))
  (define tag2 (string-append "</" tag ">"))
  `(map cadr (regex-find-groups
    ,str
    ,(string-append
      tag1
      "([\\s\\S]*?)"
      tag2))))

(define feed-url
  "https://youtube.com/feeds/videos.xml?channel_id=UCCPlzrYmF2rE34KHcDdql6A")

(define feed (request "GET" feed-url))

(zip (find-xml-tag feed "media:title")
    (find-xml-tag feed "media:description"))

```

Рисунок 2 – Просмотр публикации новых видео на Youtube канале при помощи RSS

При помощи этого скрипта можно читать публикацию новых видео на Youtube канале, указанном в переменной feed-url, при помощи RSS.

ЗАКЛЮЧЕНИЕ

В ходе проведенной работы был написан интерпретатор собственного диалекта языка Lisp, написано приложение на Android, позволяющее создавать и выполнять скрипты.

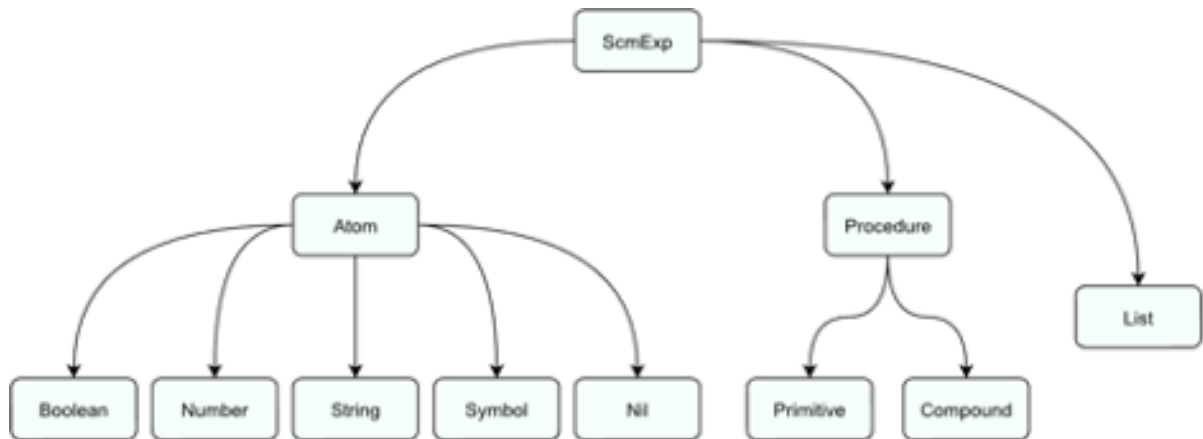
В связи с тем, что созданный язык является языком общего назначения (general-purpose), помимо агрегации новостей, данные скрипты могут быть использованы для решения любых задач.

Созданный интерпретатор позволяет проводить дальнейшие исследования языков программирования.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

Абельсон Х., Сассман Д. Интерпретация компьютерных программ: Глава 4
«Метаязыковая абстракция»

Приложение А. Иерархия типов данных языка



Приложение Б. Процесс интерпретации скрипта интерпретатором



Приложение В. Пример создания и использования макроса в языке

```
(defmacro (cond body)
  (fold nil
    (lambda (acc e)
      `(if ,(car e) ,(cdr e) ,acc))
    (reverse body)))
```

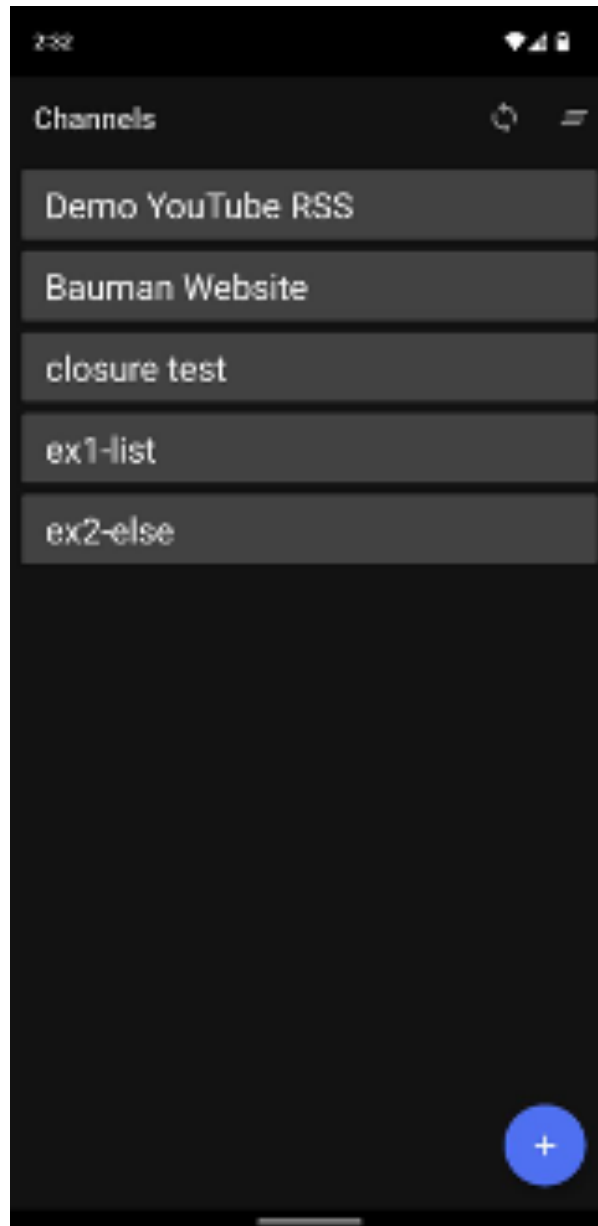
```
(cond (((= 1 2) (+ 1 2 3))
      ((= 1 1) (- 5 2))))
```

;; → 3

Приложение Г. Пример раскрытия специальных форм

<code>(define (a x y z) ...)</code>	→	<code>(define a (lambda (x y z) ...))</code>
<code>(if test a)</code>	→	<code>(if test a nil)</code>
<code>`(1 ,a 2)</code>	→	<code>(list 1 a 2)</code>
<code>(cond (((= 1 2) (+ 1 2 3)) ((= 1 1) (- 5 2))))</code>	→	<code>(if (= 1 2) (+ 1 2 3) (if (= 1 1) (- 5 2) nil))</code>

Приложение Д. Главное меню приложения



Приложение Е. Новостная лента при выполнении скрипта



Приложение Ж. Различные способы отображения результата выполнения

