# Modelos y paradigmas de programación

# Análisis del lenguaje de programación R

Harold Armando Achicanoy Estrella

**Resumen**

El objetivo de este artículo es presentar una revisión detallada a partir de los modelos y paradigmas de programación vistos durante el curso para el lenguaje de programación R. R es un lenguaje de programación y ambiente de desarrollo usado ampliamente para el análisis estadístico y la visualización de datos. El lenguaje de programación soporta los paradigmas: Orientado a objetos y Funcional. Por otro lado, no soporta los paradigmas Declarativo, Concurrente Declarativo, Con Estado, Concurrente por paso de mensajes y Relacional. Esto hace de R un lenguaje …

**Introducción**

R es un lenguaje de programación y entorno de desarrollo gratuito y de código abierto usado ampliamente para el análisis estadístico y la visualización de datos. Su surgimiento ocurre a inicios de los años 90 por dos profesores de la Universidad de Auckland en Nueva Zelanda: Ross Ihaka y Robert Gentleman. La primera versión del lenguaje fue lanzada en agosto de 1993. Sin embargo, la versión estable de R estuvo disponible desde febrero del 2000 [cita requerida]. Los orígenes de R provienen del lenguaje de programación S el cual fue desarrollado en 1970 por John Chambers y otros en los laboratorios Bell cuyo fin también estuvo destinado al análisis estadístico de datos.

Al estar directamente influenciado por S y Lisp-Stat, estos ambientes han alentado a los estadísticos a convertirse en programadores de tal manera que ellos puedan tener un control más efectivo sobre el proceso de análisis de datos [cita requerida]. R permite la integración con otros lenguajes de programación, como C/C++, Java, Python, entre otros. Es un lenguaje de código abierto y puede ser usado prácticamente en cualquier plataforma y sistema operativo tales como Unix, Linux, Windows o Mac OS. Hoy en día, R es el lenguaje de programación para análisis estadístico más popular y es usado por gigantes de la industria como Facebook y Google [cita requerida].

R es un lenguaje de programación imperativo, lo cual indica que tiene estado explícito a diferencia del paradigma de programación declarativo.

A continuación se describe el contenido del documento en función de las secciones a desarrollar: en primera medida se presenta la sintaxis, la semántica y los tipos básicos de objetos usados en R, posteriormente se muestra una revisión de los principales paradigmas de programación implementados en R así como sus respectivas pruebas. Finalmente, se presentan las conclusiones obtenidas del análisis del lenguaje.

**Sintaxis, semántica y tipos básicos de objetos usados en R**

Tomar el paper como referencia

Revisión de los paradigmas de programación implementados en R

Tomar el paper como referencia y obtener resultados en consola

1. Funcional
2. Dinámico
3. Orientado a objetos

**Conclusiones**

**Referencias**

La sintaxis del lenguaje tiene una similaridad superficial con C, pero la semántica es de una variedad de lenguajes de programación funcional con fuertes afinidades con Lisp y APL. En particular, nos permite computar sobre el lenguaje

Evaluación perezosa

Vectores: celdas continuas que contienen datos. Las celdas se acceden a través de operaciones de indexación.

Listas: vectores genéricos.

Objetos del lenguaje:

Calls

Expressions

Names

Programación orientada a objetos

Programación funcional

lexical scoping borrowed from Scheme and garbage collection

the basic data type is the vector

It is functional in that functions are first-class values and arguments are passed by deep copy.

R uses lazy evaluation by default for all arguments, thus it has a pure functional core

Yet R does not optimize recursion, and instead encourages vectorized operations.

Functions are lexically scoped and their local variables can be updated, allowing for an imperative programming style.

Finally, the language supports objects.

R is decidedly single threaded, its semantics has no provisions for concurrency, and its implementation is hopelessly non-thread safe.

The basic data type in R is the vector, an ordered collection of values of the same kind.

These values can be either numerics, characters, or logicals.

Other data types include lists (i.e., heterogeneous vectors) and functions

Arithmetic operations on vectors are performed element by element, and shorter vectors are automatically extended to match longer ones by an implicit extension process

In R, computation happens by evaluating functions.

Functions are first class values that can be created, stored and invoked

R is lazy, thus evaluation of function arguments is delayed.

Alcance dinámico

Cuando se busca c en la llamada a la función, se omite el enlace porque 42 no es una función, y en cambio la búsqueda devuelve la definición predeterminada de c, que de hecho es una función.

las reglas de búsqueda intentan encontrar una función cuando se está en un contexto de llamada, lo que puede arrojar resultados sorprendentes.