



Eficiencia de alto nivel Un acercamiento al lenguaje de programación Julia

Dr. Oscar Alejandro Esquivel Flores¹, Óscar Anuar Alvarado Morán², and Mario
Horacio Garrido Czacki²

¹Departamento de Ingeniería en Sistemas Computacionales y Automatización , Instituto de
Investigaciones en Matemáticas Aplicadas y en Sistemas, Universidad Nacional Autónoma de México

²Lic. en Ciencia de Datos, Instituto de Matemáticas Aplicadas y en Sistemas, Universidad Nacional
Autónoma de México

29 de julio de 2021

PRESENTACIÓN

El lenguaje de programación Julia es un lenguaje de alto nivel orientado hacia el cómputo científico. Posee una sintaxis muy familiar a otros lenguajes de alto nivel como Python o Pearl, desempeño se acerca a los lenguajes más potentes como C y/o Fortran. Es una alternativa libre, expresiva, amigable y veloz para implementar y ejecutar algoritmos que demandan alta carga computacional. Julia ofrece muchas ventajas sobre otros lenguajes de alto nivel como los son: un conjunto de declaraciones de tipos extenso y variado, el entorno de compilación LLVM de código nativo, despacho múltiple, paralelismo, entre muchas otras. Este lenguaje fue diseñado para hacer sencilla la codificación al mismo tiempo que logra incrementar la velocidad en la ejecución del código. En poco tiempo Julia se ha posicionado como una herramienta ideal para científicos, tecnólogos, estudiantes y docentes interesados en desarrollar aplicaciones dirigidas al cómputo científico y de alto rendimiento.

Índice

1. Introducción	1
1.1. Objetivo	1
1.2. Contenidos	1
1.3. Pre-requisitos	1
2. Instalación de Julia	1
3. Referencias	2

1. Introducción

En este curso-taller se presentará y estudiará el lenguaje de programación Julia y sus características principales. Por medio de la elaboración de código en Julia se resaltarán sus características principales como son la claridad de su sintaxis, eficiencia de ejecución y un conjunto de herramientas diversas que posee. Durante las 20 horas del curso, repartidas en 5 días de 4 hrs., los participantes conocerán los principios básicos del lenguaje y sus excelentes capacidades para desarrollar código expresivo y eficiente.

1.1. Objetivo

El objetivo general del curso-taller consiste en que el participante conozca las ventajas computacionales de este lenguaje de programación y desarrolle habilidades para elaborar código eficiente en poco tiempo con lo que podrá evaluar y valorar el lenguaje como una alternativa para desarrollar algoritmos de alto rendimiento.

1.2. Contenidos

El curso abarcará los siguientes temas:

- Presentación de Julia
- Elementos básicos
- Despacho múltiple y rendimiento
- Ecosistema de Julia
- Introducción al paralelismo
- Caso de aplicación

1.3. Pre-requisitos

Se valora, aunque no es indispensable, el conocimiento de algún lenguaje de programación de alto nivel como Python, R y/o Matlab, el conocimiento de la tecnología Jupyter notebooks se agradecerá bastante. Se recomienda el uso de un sistema operativo tipo Unix, *linux*, *MacOS*, en Windows... bueno, también se puede trabajar.

2. Instalación de Julia

Para adelantar las tareas planteadas en la sección de instalación, se propone consultar la página oficial de Julia <https://julialang.org/> y descargar la versión 1.6 del lenguaje. Julia está disponible en su versión más reciente y estable para distintos sistemas operativos en <https://julialang.org/downloads/>. Se recomienda instalación de los archivos binarios para la ejecución de Julia con la línea de comandos. Adicionalmente se instalará el *kernel* o núcleo de Jupyter Notebook (<https://jupyter.org/>) para Julia con la finalidad de ejecutar el código en este ambiente de desarrollo.

Agradecimientos

Agredecemos el apoyo brindado por la dirección general del IIMAS a través de la secretaría académica, así como a la coordinación de la licenciatura en ciencia de datos y del posgrado en ciencia e ingeniería de la computación. Especial agradecimiento merece Programa de Apoyo a Proyectos de Investigación e Innovación Tecnológica PAPIIT por financiar el proyecto IA104720 de surge este curso-taller y otros productos de investigación.

3. Referencias

- [1] Allen Downey, Ben Lauwens, *Think Julia: How to Think Like a Computer Scientist*, O'Reilly, 2018.
- [2] Changhyun Kwon, *Julia Programming for Operations Research*, Second edition, University of South Florida, 2011.
- [3] Paul D. McNicholas, Peter Tait, *Data Science with Julia*, 1st Edition, CRC Press, 2019.
- [4] Bogumil Kaminski, Przemyslaw Szufel, *Julia 1.0 Programming Cookbook*, Packt Publishing, 2018.
- [5] Avik Sengupta, *Julia High Performance*, Second edition, Packt Publishing, 2019.
- [6] Zacharias Voulgaris, *Julia for data Science*, First Edition, Technics Publications, 2016.
- [7] Okten, Giray, *First Semester in Numerical Analysis with Julia*, DigiNole: FSU'S Digital Repository, 2019.