

Algoritmos Computacionales

Grupo 3009

Profesor: Aldo Sayeg Pasos Trejo
Ayudante: David Leonardo Galicia Praskauer

31 de enero de 2020

1. Introducción

La finalidad del curso de “Algoritmos Computacionales” es introducir a los alumnos a la computación y, más principalmente, a la programación aplicada a la ciencias naturales. La clase no se enfocará en el uso de la computadora para auxiliar a los problemas técnicos de las ciencias naturales (como la edición de texto, el análisis de datos, entre otros) si no como herramienta para solucionar problemas matemáticos de manera numérica. Debido a que la computadora será nuestra herramienta fundamental. Estar familiarizado con el uso básico de la computadora ayudará al alumno durante el curso.

Por ser de naturaleza muy similar, se le enseñará a los alumnos a programar en los lenguajes Julia y Python. Consideramos que estos son los lenguajes más aptos, el primero por su excelente diseño y desempeño y el segundo por su sencillez y uso en todo tipo de aplicaciones.

2. Temario

1. Introducción a la programación y a las ciencias de la computación- (26 horas)
 - 1.1 Introducción histórica y repaso de lógica proposicional.

- 1.2 Composición de una computadora y representación de la información.
- 1.3 Estructuras de control, ciclos y programación.
- 1.4 Pseudocódigo y diagramas de flujo.
- 1.5 Introducción a los algoritmos y complejidad computacional.
- 1.6 Introducción a los lenguajes Julia y Python.
2. Introducción a la Física Computacional (28 horas)
 - 2.1 Cálculos directos.
 - 2.2 Gráficas en \mathbb{R}^2 y \mathbb{R}^3 .
 - 2.3 Puntos fijos y solución de ecuaciones trascendentes.
 - 2.4 Derivación e integración.
 - 2.5 Ecuaciones diferenciales ordinarias.
3. Temas Selectos (10 horas)
 - 3.1 Programación modular y almacenamiento de datos.
 - 3.2 Introducción a los lenguajes C y C++.
 - 3.3 Captura de datos en tiempo real mediante Arduino.
 - 3.4 Introducción a Mathematica.

3. Evaluación

Los porcentajes de evaluación del curso serán los siguientes:

- 80 % Tareas
- 20 % Examen intermedio

Las tareas serán cortas y se dejarán con una frecuencia casi semanal, dejando al menos cinco días entre la asignación y la fecha de entrega. Tanto la asignación como la entrega de las tareas se harán mediante un blog en la plataforma Google Classroom, por lo que es indispensable que el alumno cuente con un correo de dominio `@ciencias.unam.mx`. Es responsabilidad del alumno conseguir acceso al blog y estar al pendiente de las asignaciones.

El examen intermedio se realizará entre la sexta y la décima semana de clases, al concluir con el bloque de “Introducción a la programación a las ciencias de computación”. La finalidad del examen es confirmar que el alumno es capaz de programar y que entiende la estructura básica de un algoritmo. La fecha del examen será acordada en clase. No habrá reposición del examen.

Aunque la asistencia al curso no es obligatoria, la participación y el involucramiento en tanto en clase como en las tareas y problemas del curso puede ayudar a su calificación a la hora de hacer las calificaciones y evaluaciones finales.

4. Bibliografía

4.1. Básica

- Behrouz A Forouzan and Firouz Mosharraf. *Foundations of Computer Science*. Cengage Learning EMEA, 2007
- Svein Linge and Hans Petter Langtangen. *Programming for Computations – Python*. Springer, 2016
- John M Zelle. *Python programming: an Introduction to Computer Science*. Franklin, Beedle & Associates, Inc., 2004
- Christian Hill. *Learning Scientific Programming with Python*. Cambridge University Press, USA, 1st edition, 2016
- Allen Downey and Ben Lauwens. Think julia: How to think like a computer scientist. <https://benlauwens.github.io/ThinkJulia.jl/latest/book.html>
- Julia language documentation. <https://docs.julialang.org/en/v1/>
- Mark Newman. *Computational Physics*. CreateSpace Independent Publ., 2013

4.2. Complementaria

- Rubin H. Landau, Manuel J. Páez, and Cristian C. Bordeianu. *Computational Physics: Problem Solving with Python*. Wiley-VCH, 3rd edition, 2015
- John H Mathews, Kurtis D Fink, et al. *Numerical methods using MATLAB*, volume 4. Pearson Prentice Hall Upper Saddle River, NJ, 2004
- Thomas H Cormen, Charles E Leiserson, Ronald L Rivest, and Clifford Stein. *Introduction to Algorithms*. MIT press, third edition, 2009
- Subrata Dasgupta. *Computer Science: a very short introduction*, volume 466. Oxford University Press, 2016
- Martin Davis. *The Universal Computer: the road from Leibniz to Turing*. AK Peters/CRC Press, 2011
- E. Nagel, J.R. Newman, and D.R. Hofstadter. *Godel's Proof*. NYU Press, 2001
- R. Penrose. *La mente nueva del emperador: en torno a la cibernética, la mente y las leyes de la física*. Ciencia y Tecnología Series. Fondo de Cultura Económica, 2005