Universidad Nacional Autónoma de México

Facultad de Ciencias

Reporte de Actividad Docente

Manual de ejercicios para la materia de Lenguajes de Programación.

Que para obtener el Títutlo de:

Licenciado en Ciencias de la Computación.

Presenta: Luis Miguel Muñoz Barón

Tutor: Javier Enriquez Mendoza

2023

Nota al lector

El presente trabajo es una compilación de ejercicios teóricos para la materia de lenguajes de programación de la carrera en ciencias de la computación con clave de asignatura 1536, siguiendo el plan de estudios impartido desde el año 2013 en la Facultad de Ciencias de la Universidad Nacional Autónoma de México.

El objetivo principal de este material es que sirva como guía para desarrollar los contenidos visitados en este curso siendo un apoyo didáctico para profesores y alumnos donde cada ejercicio planteado durante el capítulo esté acompañado de la base teórica para poder seguir el desarrollo del mismo y ser entendido en su totalidad.

Cada capítulo cuenta con dos secciones: una sección de teoría autocontenida con ejercicios resueltos y una sección de ejercicios sin respuesta para el lector al final del mismo.

Las demostraciones y desarrollo de los contenidos teóricos serán discutidos de forma laxa dejando a discreción del lector el estudio a profundidad de la información aquí presentada concentrándonos únicamente en la teoría necesaria para poder dar solución a los ejercicios, asimilar los conceptos y definiciones, familiarizarnos con la explicación, planteamiento y respuesta del material aquí presentado.

Para un estudio en profundidad se refiere a las personas interesadas a consultar [1], [2], [5], [12], [22] y [26] de la bibliografía que se encuentra al final del presente manual.

Finalmente espero que las personas que hagan uso de este material encuentren útil y fructífero los desarrollos y conceptos escritos en las páginas de este texto.

Miguel Barón.

"A language is not just words. It's a culture, a tradition, a unification of a community, a whole history that creates what a community is. It's all embodied in a language"

-Noam Chomsky

Índice general

1. Introducción **1**

2. Herramientas matemáticas **2**

3. Sintaxis **3**

4. Semántica **4**

5. Cálculo Lambda **5**

6. MinHs **6**

7. Inferencia de tipos **7**

8. Máquinas abstractas **8**

9. TinyC **9**

10. Herencia y subtipos **10**

11. Java Peso Pluma **11**

Capítulo 1

Introducción

Capítulo 2

Herramientas matemáticas

Capítulo 3

Sintaxis

Capítulo 4

Semántica

Capítulo 5

Cálculo Lambda

Capítulo 6

MinHs

Capítulo 7

Inferencia de tipos

Capítulo 8

Máquinas abstractas

Capítulo 9

TinyC

Capítulo 10

Herencia y subtipos

Capítulo 11

Java Peso Pluma

Bibliografía

[1] Ramírez Pulido K., Soto Romero M., Enríquez Mendoza J., *Nota de Clase del curso de Lenguajes de Programación*, Facultad de Ciencias, Universidad Nacional Autónoma de México, Ciudad de México, 2022.

[2] Brooks A., *Modern Programming Languages: A Practical Introduction* (2nd Edition). Franklin, Beedle & Associates, cop. Sherwood, Oregon, 2011.

[3] Deepika P., *Selection Sort* (2021), [Fecha de consulta: 14/11/2022]. Geeks for geeks. Disponible en https://www.geeksforgeeks.org/selection-sort/

[4] Lipovaca M., *Learn You a Haskell for Great Good!: A Beginner's Guide.* (digital publication). San Francisco, California. 2011.

[5] Miranda Perea F., González Huesca L., *Nota de Clase del curso de Lenguajes de Programación,* Facultad de Ciencias, Universidad Nacional Autónoma de México, Ciudad de México, 2021.

[6] Keller G., O'Connor-Davis L., *Class Notes from the course Concepts of programming language design*, Department of Information and Computing Sciences, Utrecht University, The Netherlands, 2020.

[7] Nielson F., *Semantics with Applications: An Appetizer*, Springer Publishing, 2007.

[8] Harper R., *Practical Foundations for Programming Languages.* Working draft, Carnegie Mellon University Press, San Francisco, California, 2010. Disponible en https://moss.cs.iit.edu/cs440/readings/harper.pdf

[9] Mitchell J., *Foundations for Programming Languages,* Massachusetts Institute of Technology Press, Cambridge, Massachusetts, 1996.

[10] Krishnamurthi S., *Programming Languages Application and Interpretation,* Brown University press, Providence, Rhode Island, 2007.

[11] Spector-Zabusky A., *How would the Lambda Calculus add numbers?* (2021) [fecha de consulta: 16/4/2023.]. Stack Overflow. Disponible en https://stackoverflow.com/questions/29756732/how-would-the-lambda-calculus-add-numbers

[12] Enríquez Mendoza J., *Lenguajes de Programación Nota de clase.* Facultad de Ciencias, Universidad Nacional Autónoma de México, Ciudad de México, 2022.

[13] Keller. G., O'Connor-Davis. L., *Concepts of Programming Languages: Data types in Explicitly Typed Lenguages,* Department of Information and Computing Sciences, Utrecht University, The Netherlands, 2022.

[14] Karavirta V., Shaffer C., *Formal Languages Spring Chapter 1 Introduction Grammar Exercises* (online tool), [fecha de consulta: 7/10/2022]. Disponible en https://opendsa-server.cs.vt.edu/OpenDSA/Books/PIFLAS21/html/IntroGrammarEx.html

[15] JerrettDavis, *Binary Search, a haskell approach* (2022), [fecha de consulta: 29/11/2022]. Disponible en https://programming-idioms.org/idiom/124/binary-search-for-a-value-in-sorted-array/2120/haskell

[16] kirankumarambati, *Binary Search Data Structure and Algorithm Tutorials* (2023), [fecha de consulta: 29/11/2022]. Geeks for geeks. Disponible en https://www.geeksforgeeks.org/binary-search/

[17] Dahiya A., Wasserman Z., *CIS 194: Introduction to Haskell, Homework 1* (2013), [fecha de consulta: 4/11/2022]. University of Pennsylvania, Filadelfia, Pensilvania. Disponible en https://www.seas.upenn.edu/ cis1940/spring13/hw/01-intro.pdf

[18] Yorgey B., *Typeclassopedia* (2011), [fecha de consulta: 24/11/2022]. Wiki Haskell. Disponible en https://wiki.haskell.org/Typeclassopedia

[19] King J. K., *Haskell List Problem Set* (2018), [fecha de consulta: 10/12/22]. Github. Disponible en https://github.com/JD95/haskell-problem-sets/blob/master/Lists/Problems.hs

[20] Goguen J. A., *Semantics of computation. Category Theory Applied to Computation and Control. Lecture Notes in Computer Science*. Vol. 25. Springer, 1975.

[21] Floyd, R. W., *Assigning Meanings to Programs. In Schwartz, J.T. (ed.). Mathematical Aspects of Computer Science. Proceedings of Symposium on Applied Mathematics*. Vol. 19. American Mathematical Society, 1967.

[22] Winskel G. *The formal semantics of programming languages: an introduction*, Massachussetts Institute of Technology Press, Cambridge, Massachussetts, 1993.

[23] Schmidt, D. A., *Denotational Semantics: A Methodology for Language Development*. William C. Brown Publishers, 1986.

[24] Plotkin, G. D., *A structural approach to operational semantics* (Technical Report DAIMI FN-19), Computer Science Department, Aarhus University, Denmark, 1981.

[25] Deransart, P., Jourdan M., Lorho B., *Attribute Grammars: Definitions, Systems and Bibliography* (Lecture Notes in Computer Science 323), Springer-Verlag, Berlin Heidelberg, 1988.

[26] Krishnamurthi S., *Programming Languages: Application and Interpretation* (2nd ed.), Brown University Press, Providence, Rhode Island, 2012.

[27] Slonneger K., Kurtz B. L., *Formal Syntax and Semantics of Programming Languages*, Addison-Wesley Publishing Co., United States, 1995.

[28] Colaboradores de Wikipedia, *Semantics (computer science)*, Wikipedia, La enciclopedia libre (2023), [fecha de consulta: 18/10/2023]. Disponible en https://en.wikipedia.org/wiki/Semantics_(computer_science)

[29] Colaboradores de Wikipedia, *Syntax (programming languages)*, Wikipedia, La enciclopedia libre (2023), [fecha de consulta: 18/10/2023]. Disponible en https://en.wikipedia.org/wiki/Syntax_(programming_languages)

[30] Friedman D. P., Mitchell W., Haynes C. T., *Essentials of Programming Languages* (1st ed.), The Massachusetts Insitute of Technology Press, Cambridge, Massachusetts, 1992.

[31] Smith D., *Designing Maintainable Software*. Springer Science & Business Media, United States, 1999.

[32] Aho A. V., Lam M. S., Seth R., Ullman J. D., *Compilers: Principles, Techniques, and Tools* (2nd ed.). Addison Wesley Publishing Co., United States, 2017

[33] Louden K. C., *Compiler Construction: Principles and Practice*. Brooks-Cole Publishers. United States, 1997.

[34] Sipser M., *Introduction to the Theory of Computation*. PWS Publishing Co., United States, 1997.

[35] Colaboradores de Wikipedia. *Pragmatics*, Wikipedia, La enciclopedia libre (2023), [fecha de consulta: 19 10 2023]. Disponible en https://en.wikipedia.org/wiki/Pragmatics

[36] Coppock E., Champollion L., *Invitation to Formal Semantics (manuscript draft)*, eClass NKUA digital plataform, 2019.

[37] Mey J. L., *Pragmatics: An Introduction* (2nd ed.). Oxford-Blackwell Publishing, United Kingdom, 2001.

[38] Winter Y., *Flexibility principles in Boolean semantics*. Massachusetts Institute of Technology Press, Cambridge, Massachusetts, 2001.

[39] Felleisen M., Findler R. B., Flatt M., Krishnamurthi S., *How to Design Programs* (1st ed.), Massachusetts Institute of Technology Press, Cambridge, Massachusetts, 2003.

[40] Colaboradores de Wikipedia. *Ambiguous grammar*, Wikipedia, La enciclopedia libre (2023), [fecha de consulta: 19/10/2023]. Disponible en https://en.wikipedia.org/wiki/Ambiguous_grammar

[41] Levelt W., *An Introduction to the Theory of Formal Languages and Automata*. John Benjamins Publishing. United States, 2008.

[42] Scott E., *SPPF-Style Parsing From Earley Recognizers* (Electronic Notes in Theoretical Computer Science). Elsevier B.V. Royal Holloway, University of London Egham, Surrey, United Kingdom, 2008.

[43] Colaboradores de Wikipedia. *Compiled Language*. Wikipedia, La enciclopedia libre (2023), [fecha de consulta: 19 10 2023]. Disponible en https://en.wikipedia.org/wiki/Compiled_language

[44] Colaboradores de Wikipedia. *Interpreter (computing)*, Wikipedia, La enciclopedia libre (2023), [fecha de consulta: 19 10 2023]. Disponible en https://en.wikipedia.org/wiki/Interpreter_(computing)

[45] Terence P., Luber J., *The Difference Between Compilers and Interpreters*, Wayback Machine Archive, United States, 2014.

[46] Colaboradores de IONOS Digital Guide, *Compilers vs. interpreters: explanation and differences*, IONOS Digital Guide (2023), [fecha de consulta: 26/11/2023]. Disponible en: https://www.ionos.com/digitalguide/websites/web-development/compilers-vs-interpreters

[47] Colaboradores de Wikipedia. *Object-oriented programming*. Wikipedia, La enciclopedia libre (2023), [fecha de consulta: 19 10 2023]. Disponible en https://en.wikipedia.org/wiki/Object-oriented_programming

[48] Martin A., Cardelli L., *A Theory of Objects*, Springer Verlag. United States, 1998.

[49] Armstrong D. J., *The Quarks of Object-Oriented Development* (Communications of the ACM), Research Gate digital archive, 2006. Disponible en: https://www.researchgate.net/publication/220425366_The_quarks_of_object-oriented_development

[50] Colaboradores de Wikipedia. *Programación por procedimientos*, Wikipedia, La enciclopedia libre (2023), [fecha de consulta: 19 10 2023]. Disponible en https://es.wikipedia.org/wiki/Programacion_por_procedimientos

[51] Colaboradores de Wikipedia. *Functional programming*. Wikipedia, La enciclopedia libre (2023), [fecha de consulta: 23 10 2023]. Disponible en https://en.wikipedia.org/wiki/Functional_programming

[52] Hudak P., *Conception, evolution, and application of functional programming languages*. ACM Computing Surveys. Yale University, Department of Computer Science, New Haven, Connecticut, 1989.

[53] Jain A., *Javascript Promises: Is There a Better Approach?* Medium (2023), [fecha de consulta: 29/11/2023]. Disponible en: https://medium.datadriveninvestor.com/javascript-promises-is-there-a-better-approach

[54] Colaboradores de Wikipedia, *Imperative programming*. Wikipedia, La enciclopedia libre (2023). [fecha de consulta: 23/10/2023]. Disponible en https://en.wikipedia.org/wiki/Imperative_programming

[55] Colaboradores de IONOS Digital Guide, *Imperative programming: Overview of the oldest programming paradigm*. IONOS Digital Guide (2021), [fecha de consulta: 21/4/2022]. Disponible en: https://www.ionos.com/digitalguide/websites/web-development/imperative-programming/

[56] Eckel B., *Thinking in Java*, Pearson Education Publishers. United States, 2006.

[57] Colaboradores de Wikipedia. *Logic programming*, Wikipedia, La enciclopedia libre (2023), [fecha de consulta: 24/10/2023]. Disponible en https://en.wikipedia.org/wiki/Logic_programming

[58] Colaboradores de Wikipedia. *Mathematical object*, Wikipedia, La enciclopedia libre (2023), [fecha de consulta: 25/10/2023]. Disponible en https://en.wikipedia.org/wiki/Mathematical_object

[59] Azzouni, J., *Metaphysical Myths, Mathematical Practice*, Cambridge University Press, United States, 1994.

[60] Burgess J., Rosen G., *A Subject with No Object*, Oxford University Press, United Kingdom, 1997.

[61] Colaboradores de Wikipedia. *Judgment (mathematical logic)*, Wikipedia, La enciclopedia libre (2023), [fecha de consulta: 25/10/2023]. Disponible en https://en.wikipedia.org/wiki/Judgment_(mathematical_logic)

[62] Martin-Löf P., *On the meanings of the logical constants and the justifications of the logical laws*. Nordic Journal of Philosophical Logic, Department of Mathematics, University of Stockholm, Sweden, 1996.

[63] Colaboradores de Wikipedia. *Rule of inference*, Wikipedia, La enciclopedia libre (2023), [fecha de consulta: 25/10/2023]. Disponible en https://en.wikipedia.org/wiki/Rule_of_inference

[64] Boolos G., Burgess J., Jeffrey R. C., *Computability and logic*, Cambridge University Press, United States, 2007.

[65] John C. R., *Theories of Programming Languages*, Cambridge University Press, United States, 2009.

[66] Bergmann M., *An introduction to many-valued and fuzzy logic: semantics, algebras, and derivation systems*, Cambridge University Press, United States, 2008.

[67] Miranda F., Viso E. G., *Matemáticas Discretas*, Facultad de Ciencias, Universidad Nacional Autónoma de México, Ciudad de México, 2016.

[68] Dossey J. A., Otto D. A., Spence L. E., Eynden C. V., *Discrete Mathematics* (5-th edition), Pearson-Addison-Wesley Publishing Co., Boston, United States, 2006.

[69] Gersting J. L., *Mathematical Structures for Computer Science* (3rd edition), Computer Science Press, W.H. Freeman and Company, United States, 1993.

[70] Grassman W. K., Tremblay J., *Logic and Discrete Mathematics, A computer Science Perspective*, Prentice-Hall Inc., United States, 1996.

[71] Gries D., Schneider F. B. *A Logical Approach to Discrete Mathematics*, Springer-Verlag, United States, 1994.

[72] Grossman J. W., *Discrete Mathematics, An introduction to concepts, methods and applications*, Macmillan Publishing Company, United States, 1990.

[73] Koshy T., *Discrete Mathematics with Applications*, Elsevier Academic Press, 2004.

[74] Rossen K. H., *Discrete Mathematics and its Applications* (6-th edition), McGraw Hill, 2006.

[75] Jessica S., *Introduction to Haskell* (2013), [fecha de consulta: 4/11/2022]. University of Pennsylvania, Filadelfia, Pennsylvania. Disponible en https://www.seas.upenn.edu/ cis1940

[76] Krahn, H., Rumpe B., Völke S., *Model Driven Engineering Languages and Systems*. Technische Universität Braunschweig, Braunschweig, Germany. 2007.

[77] Chomsky N. *Aspects of the Theory of Syntax*, Massachusetts Institute of Technology Press, Cambridge, Massachusetts, 2014.

[78] Colaboradores de Wikipedia, *Parse tree*, Wikipedia, La enciclopedia libre (2023), [fecha de consulta: 30/10/2023]. Disponible en https://en.wikipedia.org/wiki/Parse_tree

[79] Colaboradores de Wikipedia, *Abstract syntax*, Wikipedia, La enciclopedia libre (2023), [fecha de consulta: 30/10/2023]. Disponible en https://en.wikipedia.org/wiki/Abstract_syntax

[80] Colaboradores de Wikipedia, *Scope*, Wikipedia, La enciclopedia libre (2023), [fecha de consulta: 31/10/2023]. Disponible en https://en.wikipedia.org/wiki/Scope_(computer_science)

[81] Colaboradores de Wikipedia, *Let expression*, Wikipedia, La enciclopedia libre (2023). [fecha de consulta: 31 10 2023]. Disponible en https://en.wikipedia.org/wiki/Let_expression

[82] Colaboradores de Wikipedia, *Lamabda Calculus*, Wikipedia, La enciclopedia libre (2023), [fecha de consulta: 31 10 2023]. Disponible en https://en.wikipedia.org/wiki/Lambda_calculus

[83] Turing, A., *Computability and λ-Definability*, The Journal of Symbolic Logic, United Kingdom, 1937.

[84] Thierry C., Zalta E. N., *Type Theory*, The Stanford Encyclopedia of Philosophy, Department of Philosophy, Stanford University, United States, 2013.

[85] Mitchell J. C., *Concepts in Programming Languages*, Cambridge University Press. Cambridge, Massachusetts, United States, 2003.

[86] Pierce B. C., *Basic Category Theory for Computer Scientists*, The MIT Press, Cambridge, Massachusetts, 1991.

[87] Church A., *A set of postulates for the foundation of logic*, (Annals of Mathematics Archive), Mathematics Department, Princeton University Press, Princeton, Nueva Jersey, 1932.

[88] Selinger P., *Lecture Notes on the Lambda Calculus* (vol. 0804), Department of Mathematics and Statistics, University of Ottawa Press, Ottawa, Canada, 2018.

[89] Turbak F., Gifford D., *Design concepts in programming languages*, The MIT press, Cambridge, Massachusetts, 2008.

[90] Abrahams, P. W., *A final solution to the Dangling else of ALGOL 60 and related languages*, Communications of the ACM, Volume 9, Issue 9, 1986.

[91] Colaboradores de Wikipedia. *Mathematical induction*, Wikipedia, La enciclopedia libre (2023), [fecha de consulta: 14 11 2023]. Disponible en https://en.wikipedia.org/wiki/Mathematical_induction

[92] DeVos M., *Mathematical Induction*, Simon Fraser University Press, British Columbia, Canada, 2023.

[93] Diaz G., *Mathematical Induction* (Wayback Machine Archive), Harvard University Press, Cambridge, Massachusetts, 2023.

[94] Colaboradores de Wikipedia. *Recursion*, Wikipedia, La enciclopedia libre (2023), [fecha de consulta: 14 11 2023]. Disponible en https://en.wikipedia.org/wiki/Recursion

[95] Causey R. L., *Logic, sets, and recursion* (2nd ed.), Sudbury, Mass: Jones and Bartlett Publishers, New England, 2006.

[96] user207421. *Static Semantics meaning?* Stack Overflow (2016), [fecha de consulta: 14/11/2023], Disponible en: https://stackoverflow.com/questions/40430578/static-semantics-meaning

[97] Remer F., *Compiler Construction course*, University of California Press, Santz Cruz, 1979.

[98] Colaboradores de Wikipedia. *Dynamic syntax*. Wikipedia, La enciclopedia libre, 2023 [fecha de consulta: 28 11 2023]. Disponible en https://en.wikipedia.org/wiki/Dynamic_syntax

[99] Cann R., Kempson R., Lutz M., *The dynamics of language: an introduction*, Elsevier, Amsterdam, 2005.

[100] Colaboradores de Wikipedia. *Operational semantics*, Wikipedia, La enciclopedia libre (2023), [fecha de consulta: 28 11 2023]. Disponible en https://en.wikipedia.org/wiki/Operational_semantics

[101] Gilles K. *Natural Semantics*, Proceedings of the 4th Annual Symposium on Theoretical Aspects of Computer Science, Springer-Verlag, London, 1987.

[102]