PROGRAMA de Teoría de la Computación

Carrera: Licenciatura en Informática

Asignatura: Teoría de la Computación

Núcleo al que pertenece: Avanzado

Profesores: Pablo Factorovich y Pablo Barenbaum

Asignaturas Correlativas: Lenguajes Formales y Autómatas y Algoritmos

Objetivos:

- Que les alumnes comprendan qué es un modelo de cómputo y la equivalencia entre cualquier par de modelos de computo de propósito general (Tesis de Church-Turing) en cuanto a la resolución de problemas.
- Que les alumnes entiendan los límites de la computabilidad y puedan demostrar y entender si un problema es decidible y/o reconocible (recursivamente enumerable).
- Que les alumnes puedan codificar los problemas utilizando lenguajes y entiendan qué es un decisor, qué es un enumerador y cuál es la relación intuitiva entre máquinas de Turing y algoritmos.
- Que les alumnes desarrollen la intuición de la equivalencia entre máquinas de Turing universales con los intérpretes de lenguajes y, en particular, con la arquitectura de Von Neumann.
- Que les alumnes desarrollen tanto una noción informal como formal de problema tratable vs. intratable y comprendan los límites prácticos de la computación.
- Que les alumnes diferencien entre la dificultad de resolver un problema vs. la certificación del mismo y cuál es la relación entre las máquinas de Turing determinísticas y las no-determinísticas.
- Que les alumnes conozcan las clases de complejidad temporal y espacial más comunes (Clases L, NL, P, NP, PSPACE, NPSPACE, EXPTIME), el concepto de completitud en cada clase (especialmente NL, NP y PSPACE), y sus contrapartes hard (NL-hard, NP-hard, PSAPCE-hard), y conozcan sus relaciones de inclusión conocidas y desconocidas (en particular P vs NP, L vs. NL, y P vs. EXP).
- Que les alumnes entiendan el concepto de reducción, particularmente la reducción many-to-one y la reducción de Turing, al igual que las reducciones de Karp y Cook y logspace, y puedan comprender cuándo aplicar cada una de ellas.

Contenidos mínimos:

- Máquinas de Turing. Máquinas algorítmicas.
- Problemas computables y no computables
- Problema de la parada.
- Problemas tratables e intratables.
- Conjuntos decidibles, r.e., reducciones many-one.
- Clases L, P, PSPACE, NP, NP-completitud

Carga horaria semanal: 4 hs

Programa analítico:

Unidad 0: Repaso.

Definición de demostración matemática; demostración directa, por el contrarrecíproco, por el absurdo y por inducción; terminología básica: strings, grafos, conjuntos, etc; lenguajes y codificación de problemas computacionales (1 semana)

Unidad 1: Concepto de algoritmo.

Maquinas de Turing (determinista, multicinta, no-determinista) y equivalencias entre las mismas. Enumeradores. Máquinas de Turing universales y equivalencia con los intérpretes (incluido el algoritmo fetch-decode-execute de la máquina RAM). Tesis de Church-Turing y definición de algoritmo. (3 semanas)

Unidad 2: Decidibilidad.

Definición de lenguaje decidible (computable) y reconocible (enumerable); lenguaje no decidible (lenguaje halting); lenguaje no reconocible (complemento del lenguaje halting); método de diagonalización. (2 semanas)

Unidad 3: Reducibilidad.

Ejemplos y demostraciones de lenguajes no decidibles via Turing-reducibilidad. LBAs y reducciones por historia de cómputos. Reducción many-to-one y concepto de función computable. Diferencia entre reducciones Turing y many-to-one. (2 semanas)

Unidad 4: Complejidad Temporal.

Complejidad temporal, notación O y clases de complejidad deterministicas y no-determinísticas. Clase P, clase NP, y equivalencia entre verificación polinomial y tiempo polinomial en máquina no determinística. Pregunta P = NP? NP-completitud y strongly NP-completitud. Teorema de Cook-Levin. Reducciones a la Cook y a la Karp. Métodos de demostración de NP-completitud. (4 semanas)

Unidad 5: Complejidad Espacial.

Definición y clases de complejidad determinísticas y no-determinísticas. Teorema de Savitch. Clases PSPACE y EXPTIME. Problemas PSPACE-completos y métodos de reducción. Clases L y NL. Problemas NL-completos y reducciones logspace. (2 semanas)

Bibliografía obligatoria:

- Michel Sipser. 2012. Introduction to the theory of computation (3rd ed.). Cengage Learning.
- Arnold L. Rosenberg. 2009. The Pillars of Computation Theory: State, Encoding, Nondeterminism. Springer Science & Business Media

Bibliografía de consulta:

- Turing. 1936. On Computable Numbers, with an Application to the Entscheidungsproblem, in Proceedings of the London Mathematical Society, Addison Wesley
- Michel Garey y David Johnson. 1979. Computers and Intractability: A Guide to the Theory of NP-completeness. W. H. Freeman.

Organización de las clases:

Clases teórico-prácticas mayormente, siguiendo las unidades temáticas en orden. Se reserva un pequeño espacio para poner en común los ejercicios resueltos por les alumnes en el transcurso de la semana. Los ejercicios se organizan en las siguientes prácticas, que se corresponden con las distintas unidades.

Práctica 1: Concepto de algoritmo.

Se busca que le alumne conozcas distintos modelos de cómputo para problemas de decisión como máquinas universales como las de *Turing* (en sus versiones determinista, multicinta, no-determinista) y *RAM* y la equivalencias entre las mismas en cuanto a poder de cómputo entre estas y otras según las tesis de Church-Turing. Asimismo, se busca que comprendan la noción de *enumerador* como solución a problemas cuya respuesta es un conjunto.

Práctica 2: Decidibilidad.

Se busca que les alumnes comprendan que hay problemas no decidibles y conozcan un primer ejemplo, que se demuestra por diagonalización. También la noción de reconocedor y que hay lenguajes que sin ser decidibles son reconocibles.

Práctica 3: Reducibilidad.

Se busca que les alumnes extiendan sus herramientas de demostración de la indecidibilidad de un lenguaje a Turing-reducibilidad y many-to-one. De esta forma y a partir del trabajo sobre la decidibilidad en problemas sobre LBAs que resultan indecidibles en máquinas de Turing, se pretende que les alumnes comprendan profundamente qué hace indecidibles a los problemas. También que a través de las reducciones many-to-one es posible demostrar que otros problemas no son reconocibles ni su complemento tampoco y comprender que caracteriza a estos problemas.

Práctica 4: Complejidad Temporal.

Se busca que les alumnes comprendan las clases de complejidad temporal (determinísticas o no) más importantes: P, NP. También trabajar las nociones de NP-completitud y strongly NP-completitud. y la aplicación de las reducciones a la Cook y a la Karp para mostrar que otros problemas son NP-completos.

Práctica 5: Complejidad Espacial.

Se busca que les alumnes comprendan las clases de complejidad espacial determinísticas y no-determinísticas más importantes: PSPACE, NPSPACE, L y NL. También las relaciones de inclusiones conocidas y desconocidas entre estas y P, NP, y EXPTIME a través del teorema de Savitch.

Modalidad de evaluación:

Los mecanismos de evaluación en modalidades libre y presencial de esta asignatura están reglamentados según los siguientes artículos del Régimen de estudios de la UNQ (Res. CS 201/18)
En la modalidad de libre, se evaluarán los contenidos de la asignatura con un examen escrito, un examen oral e instancias de evaluación similares a las realizadas en la modalidad presencial.

Universidad Nacional de Quilmes

CRONOGRAMA TENTATIVO

Semana	Tema/unidad	Actividad*				Evaluación
		Teórico	Práctico			
			Res Prob.	Lab.	Otros Especificar	
1	Repaso autómatas y presentación de máquinas de Turing de distinto tipo. Noción de problema y de algoritmo – Unidades 0 y 1	X	x			
2	Problemas recursivos. Existencia de problemas no decidibles y demostraciones por absurdo – Unidad 2	X	x			
3	Problema de la detención – Unidad 2	X	x			
4	Otros problemas recursivos no decidibles. Decidibilidad para problemas de cinta acotada – Unidad 2	X	x			
5	Reducibilidad muchos a uno – Unidad 3	X	x			
6	Clases de complejidad – Unidad 3	X	x			
7	Equivalencia entre modelos de cómputo y tesis de Church. Teorema de Rice – Unidad 3	X	x			
8	Turing-completitud, oráculos y jerarquía aritmética – Unidad 3	X	x			
9	Repaso		x			
10	Parcial		x			Parcial
11	Problemas tratables y no tratables. NP-completitud y reducciones I – Unidad 4	X	x			
12	NP-completitud y reducciones II – Unidad 4	X	x			
13	Comparación teoría de lo computable / teoría de lo tratable. Definición de complejidad espacial y clases PSPACE, NPSPACE, L y NL – Unidades 3, 4 y 5	X	x			
14	Relación entre las clases de complejidad temporal y espacial vistas – Unidades 4 y 5	X	x			

15	Repaso		x				
16	Parcial		x				Parcial

***INDIQUE CON UNA CRUZ LA MODALIDAD**

Firmado digitalmente por: SUAREZ Mariana Alejandra
Directora del Departamento de Ciencia y Tecnología
UNIVERSIDAD NACIONAL DE QUILMES