



INSTITUTO/S: Tecnología e Ingeniería

CARRERA/S: Licenciatura en Informática

MATERIA: Teoría de la computación

NOMBRE DEL RESPONSABLE DE LA ASIGNATURA: Miguel Carboni

EQUIPO DOCENTE: -----

CUATRIMESTRE: 1^{ro}

AÑO: 4^{to}

PROGRAMA N°: 34

(Aprob. Por Cons.Directivo 04/06/2022)

Instituto/s: Tecnología e Ingeniería

Carrera/s: Licenciatura en informática

Nombre de la materia: Teoría de la computación

Responsable de la asignatura y equipo docente: Miguel Carboni

Cuatrimestre y año: 1^{ro} del 4^{to} año

Carga horaria semanal: 4 hs

Programa N°: 34

Código de la materia en SIU: 777

Teoría de la computación

1. Fundamentación

La materia trata un eje fundamental de las ciencias de la computación, la computabilidad y complejidad computacional, necesario en la formación de un profesional de la informática. Asimismo, como distintos contenidos de la complejidad computacional actualmente están abiertos a distintos caminos de investigación, se pretende con la materia estimular este estudio brindando herramientas básicas y esenciales.

2. Propósitos y/u objetivos

Objetivos

Son objetivos de esta materia que los/as estudiantes:

- Comprendan qué es un modelo de cómputo y la equivalencia entre cualquier par de modelos de cómputo de propósito general (Tesis de Church-Turing) en cuanto a la resolución de problemas.
- Entiendan los límites de la computabilidad y puedan demostrar y entender si un problema es decidible y/o reconocible (recursivamente enumerable).
- 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.
- 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.
- Desarrollen tanto una noción informal como formal de problema tratable vs. intratable y comprendan los límites prácticos de la computación.
- 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.

- 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 conozca sus relaciones de inclusión conocidas y desconocidas (en particular P vs NP, L vs. NL, y P vs. EXP).
- 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.

3. Programa sintético:

Máquinas de Turing. Máquinas Algorítmicas. Conceptos Básicos de Teoría de Computabilidad y Complejidad: Problemas computables y no computables. Problema de la parada. Problemas tratables e intratables. Conjuntos decidibles, Funciones recursivas. Conjuntos recursivamente enumerables. Reducciones many-one. Clases L, P, PSPACE, NP, NP - completitud. Análisis de Algoritmos: Análisis asintótico, comportamiento en el mejor caso, caso promedio y peor caso. Notación $O()$. Balance entre tiempo y espacio en los algoritmos. Análisis de Complejidad de Algoritmos. Teoría de base de datos.

4. Programa analítico

4.1 Organización del contenido:

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. Máquinas de Turing (determinista, multicinta, no-determinista) y equivalencias entre las mismas. Enumeradores. Máquinas de Turing universales y equivalencia con los interpretes (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 determinísticas 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)

4.2 Bibliografía y recursos obligatorios:

Sipser, M. (2012). *Introduction to the theory of computation* (3rd ed.). USA, Boston: Cengage Learning.

Rosenberg, A.L. (2009). *The Pillars of Computation Theory: State, Encoding, Nondeterminism*. Springer Science & Business Media. USA, San Francisco: Editorial Board.

4.3 Bibliografía optativa:

Turing, A. M. (1936). On Computable Numbers, with an Application to the Entscheidungsproblem. Proceedings of the London Mathematical Society. <https://doi.org/10.1112/plms/s2-42.1.230>

Garey, M. y Johnson, D. (1979). *Computers and Intractability: A Guide to the Theory of NP-completeness*. USA, New York: W. H. Freeman.

5. Metodología de enseñanza:

Se propondrán clases teórico-prácticas de acuerdo a las unidades temáticas siguiendo el orden establecido en el cronograma.

Durante la clase presencial los/as estudiantes resolverán situaciones problemáticas y/o ejercicios organizados en prácticas, que se corresponden con las distintas unidades del programa de la siguiente manera:

Práctica 1: Concepto de algoritmo;

Práctica 2: Decidibilidad;

Práctica 3: Reducibilidad;

Práctica 4: Complejidad Temporal;

Práctica 5: Complejidad Espacial.

Se destinará un momento de la clase presencial para poner en común los ejercicios resueltos por los/as alumnos/as de manera individual y/o grupal.

Plan de trabajo en el campus:

El aula virtual de la materia se concibe como un espacio complementario de la clase presencial.

Se utilizará para la publicación de: las clases (escritas y/o grabadas); programa de la materia; cronograma de la materia; trabajos prácticos y artículos de interés.

También para las consultas de los/as alumnos/as, promoviendo un foro de discusión permanente.

6. Actividades de investigación y extensión (si hubiera)

No aplica.

7. Evaluación y régimen de aprobación

7.1 Aprobación de la cursada

Para aprobar la cursada y obtener la condición de regular, el régimen académico establece que debe obtenerse una nota no inferior a cuatro (4) puntos. Todas las instancias evaluativas deberán tener una instancia de recuperatorio. Podrán acceder a la administración de esta modalidad solo aquellos y aquellas estudiantes que hayan obtenido una nota inferior o igual a 6 (seis) puntos en el examen parcial.

Siempre que se realice una evaluación de carácter recuperatorio, la calificación que los/as estudiantes obtengan reemplazará la calificación obtenida en el examen que se ha recuperado y será la considerada definitiva a los efectos de la aprobación.

El/La alumno/a deberá poseer una asistencia no inferior al 75% en las clases presenciales.

En cuanto a la cursada de manera virtual se requerirá que el/la estudiante ingrese al aula virtual como mínimo una vez por semana.

7.2 Aprobación de la materia

La materia puede aprobarse por promoción, evaluación integradora, examen final o libre.

Promoción directa: tal como lo establece el art°17 del Régimen Académico, para acceder a esta modalidad, el/la estudiante deberá aprobar la cursada de la materia con una nota no inferior a siete (7) puntos, no obteniendo en ninguna de las instancias de evaluación parcial menos de seis (6) puntos, sean evaluaciones parciales o recuperatorios. El promedio estricto resultante deberá ser una nota igual o superior a siete (7) sin mediar ningún redondeo.

Evaluación integradora: tal como lo establece el art°18 del Régimen Académico, podrán acceder a esta evaluación aquellos estudiantes que hayan aprobado la cursada con una nota de entre cuatro (4) y seis (6) puntos.

La evaluación integradora tendrá lugar por única vez en el primer llamado a exámenes finales posterior al término de la cursada. Deberá tener lugar en el mismo día y horario de la cursada y será administrado, preferentemente, por el/la docente a cargo de la comisión. Se

aprobará tal instancia con una nota igual o superior a cuatro (4) puntos, significando la aprobación de la materia.

La nota obtenida se promediará con la nota de la cursada.

Examen final: Instancia destinada a quienes opten por no rendir la evaluación integradora o hayan regularizado la materia en cuatrimestres anteriores. Se evalúa la totalidad de los contenidos del programa de la materia y se aprueba con una calificación igual o superior a cuatro (4) puntos. Esta nota no se promedia con la cursada.

7.3 Criterios de calificación

La calificación de cada evaluación se determinará en la escala 0 a 10, con los siguientes valores: 0, 1, 2 y 3: insuficientes; 4 y 5 regular; 6 y 7 bueno; 8 y 9 distinguido; 10 sobresaliente.

8. Cronograma

El siguiente cronograma establece un marco de trabajo en función de los temas a abordar, su importancia y complejidad. Las actividades se conformarán de manera presencial, virtual o combinadas y se comunicarán al inicio de cursada de manera de fijar los encuentros.

Semana	Tema	Actividad
1	Repaso y máquinas de Turing de distinto tipo.	Teórica/práctica
2	Problemas no computables y demostración por absurdo.	Teórica/práctica
3	Reducibilidad	Teórica/práctica
4	Clases de complejidad.	Teórica/práctica
5	Equivalencia entre modelos de cómputo y tesis de Church	Teórica/práctica
6	Teorema de Rice	Teórica/práctica
7	Turing-completitud y oráculos	Teórica/práctica
8	Repaso	Práctica
9	1°Parcial	Práctica
10	Problemas tratables y no tratables	Teórica/práctica
11	NP-completitud y reducciones I	Teórica/práctica
12	NP-completitud y reducciones II	Teórica/práctica
13	Comparación teoría de lo computable / teoría de lo tratable	Teórica/práctica
14	Clases de complejidad y su relación	Teórica/práctica
15	2° Parcial	Práctica
16	Recuperatorios	Práctica