

CIENCIA DE LA COMPUTACIÓN

TEORÍA DE LA COMPUTACIÓN

4 CRÉDITOS



UTEC
Universidad
de Ingeniería
y Tecnología

ÍNDICE

ÍNDICE	2
UNIVERSIDAD DE INGENIERÍA Y TECNOLOGÍA	3
SILABO 2020-2	3
ASIGNATURA	5
DATOS GENERALES	5
Créditos: cuatro (4) créditos	5
Horas de teoría: dos (2) semanales	5
Horas de práctica: cuatro (4) semanales	5
Duración del período: dieciséis (16) semanas	5
Condición:	5
Modalidad:	5
Requisitos:	5
PROFESORES	5
Profesor coordinador del curso	5
INTRODUCCIÓN AL CURSO	5
OBJETIVOS	6
COMPETENCIAS	6
RESULTADOS DE APRENDIZAJE	6
TEMAS	6
PLAN DE TRABAJO	7
Metodología	7
Sesiones de teoría	7
Sesiones de práctica (laboratorio o taller)	9
SISTEMA DE EVALUACIÓN	9
REFERENCIAS BIBLIOGRÁFICAS	6

UNIVERSIDAD DE INGENIERÍA Y TECNOLOGÍA

SILABO 2020-2

1. ASIGNATURA

CS2101 - Teoría de la Computación

2. DATOS GENERALES

2.1 Créditos: cuatro (4) créditos

2.2 Horas de teoría: dos (2) semanales

2.3 Horas de práctica: cuatro (4) semanales

2.4 Duración del período: dieciséis (16) semanas

2.5 Condición:

- Obligatorio para Ciencia de la Computación

2.6 Modalidad: Virtual

2.7 Requisitos:

- CS1D02 ESTRUCTURAS DISCRETAS II

3. PROFESORES

3.1 Profesor coordinador del curso

Juan Gutiérrez Alva (jgutierrez@utec.edu.pe)

Horario de atención: previa coordinación con el profesor TP

3.2. Profesor(es) instructor(es) del curso

Juan Gutiérrez Alva (jgutierrez@utec.edu.pe)

Horario de atención: previa coordinación con el profesor

4. INTRODUCCIÓN AL CURSO

Este curso hace énfasis en los lenguajes formales, modelos de computación y computabilidad, además de incluir fundamentos de la complejidad computacional y de los problemas NP completos.

Presentaremos varios modelos de computación así como sus poderes y limitaciones. La pregunta primordial en este curso es: ¿Cuáles son las capacidades y limitaciones fundamentales de las computadoras?

Desarrollaremos 3 tipos de respuestas dependiendo del enfoque utilizado. Desde el punto de vista de la teoría de tratabilidad esta pregunta se reduce a cuáles problemas son fáciles (Clase P) y cuáles no lo son (Clase NP-completo). Desde el punto de vista de la computabilidad esta pregunta se reduce a saber cuáles problemas se pueden resolver (problemas decidibles) y

cuales no (problemas indecidibles). Desde el punto de vista de la teoría de autómatas, esta pregunta se reduce a saber qué modelos de autómatas son más poderosos que otros.

5. OBJETIVOS

Sesión 1. Aplicar los conceptos y técnicas fundamentales de la teoría de autómatas en la resolución de problemas.

Sesión 2. Aplicar los conceptos y técnicas fundamentales de la teoría de lenguajes formales en la resolución de problemas.

Sesión 3. Aplicar los conceptos y técnicas fundamentales de la tratabilidad y computabilidad en la resolución de problemas.

6. COMPETENCIAS

Las competencias que se van a trabajar en este curso son:

- a2: aplicar conocimientos de ciencias (nivel 2)
El estudiante resuelve un set de problemas donde aplica lo aprendido de manera teórica: diseñar un autómata/gramática, analizar la correctitud del autómata/gramática diseñado.
- b2: analizar información (nivel 2)
El estudiante identifica los requerimientos computacionales apropiados para el diseño de un autómata/gramática, con lo cual consigue diseñar un autómata/gramática eficiente para el problema y demostrar su correctitud
- d1: trabajar en equipo (nivel 2)
El estudiante participa y se comunica grupalmente en la resolución de problemas

7. RESULTADOS DE APRENDIZAJE

Al final del curso de Teoría de la Computación se espera:

RA1. Que el estudiante sea capaz de entender la diferencia entre un problema sin solución (problema indecidible) y un problema que si posee solución.

RA 2. Que el estudiante sea capaz de entender la diferencia entre un problema NP-difícil y un problema que posee solución polinomial.

RA 3. Que el estudiante sea capaz de identificar y resolver un problema mediante la teoría de autómatas. Además de reconocer cual es el tipo de autómata más simple que resuelve el problema

8. TEMAS

1. Introducción

- 1.1. Introducción al curso
- 1.2. Pruebas: por inducción, por contradicción, constructivas, no constructivas, principio de casillas

2. Autómatas finitos

- 2.1. Autómatas Finitos Deterministas (AFD)
- 2.2. Autómatas Finitos No Deterministas (AFN)
- 2.3. Transformación de AFN a AFD
- 2.4. Autómatas Finitos No Deterministas con transiciones vacías (AFN-e)
- 2.5. Transformación de AFN-e a AFD

3. Lenguajes regulares

- 3.1. Expresiones regulares
- 3.2. Gramáticas regulares
- 3.3. Lema de bombeo

4. Lenguajes independientes del contexto

- 4.1. Gramáticas independientes del contexto
- 4.2. Ambigüedad
- 4.3. Forma normal de Chomsky
- 4.4. El algoritmo CYK
- 4.5. Autómatas de Pila

5. Computabilidad

- 5.1. Máquinas de Turing
- 5.2. Lenguajes reconocibles y decidibles
- 5.3. Problema de la Parada
- 5.4. Reducciones

6. Tratabilidad

- 6.1. Las clases P y NP
- 6.2. NP completitud
- 6.3. Reducciones polinomiales

9. PLAN DE TRABAJO

9.1 Metodología

Las sesiones de teoría se llevan a cabo en clases magistrales con actividades que incluyen aprendizaje activo para permitir a los estudiantes internalizar los conceptos.

9.2 Sesiones de teoría

Las sesiones teóricas serán desarrolladas bajo la estructura de clase invertida, lo que significa que el estudiante es responsable por su aprendizaje y preparación para la sesión de clase. Antes de cada clase, los estudiantes tendrán asignado un problema (indicada por el docente) y sobre dicho ejercicio se desarrollará el tema de la clase.

9.3 Sesiones de práctica (laboratorio o taller)

Las sesiones prácticas/laboratorio se desarrollarán a través de una metodología activa generando el aprendizaje práctico por parte del estudiante. Las sesiones de práctica se caracterizan por el desarrollo de problemas tanto de demostración de propiedades como de diseño de nuevos algoritmos.

Para verificar que los alumnos hayan alcanzado el logro planteado para cada una de las unidades de aprendizaje, realizarán actividades que les permita aplicar los conocimientos adquiridos durante las sesiones de teoría y se les propondrá retos para que permitan evaluar el desempeño de los alumnos.

10. SISTEMA DE EVALUACIÓN

	TEORÍA	PRÁCTICA Y/O LABORATORIO
EVALUACIÓN	4 Exámenes (uno por módulo)(20%) La nota de Teoría se calcula: $(E1 + E2 + E3 + E4)/4$	1 Proyecto (p((20%) La nota de Práctica se calcula: $0.20 * P1$
	80%	20%
	100%	

Las rúbricas que permitirán medir las actividades más significativas del curso y que, además se relacionan con la evaluación de las competencias del estudiante son: [enlace](#)

11. REFERENCIAS BIBLIOGRÁFICAS

- John E. Hopcroft and Jeffrey D. Ullman. Introducción a la Teoría de Autómatas, Lenguajes y Computación. CECSA, 1993.
- Michael Sipser, Introduction to the Theory of Computation, 3rd Edition. Cengage Learning, 2012
- Teoría de la computación: lenguajes formales, autómatas y complejidad. J. Glenn Brookshear. Addison-Wesley Iberoamericana España, S.A., 1993