

CIENCIA DE LA COMPUTACIÓN

TEORÍA DE LA COMPUTACIÓN

4 CRÉDITOS



ÍNDICE

ASIGNATURA	3
DATOS GENERALES	3
Ciclo: 4	3
Créditos: cuatro (4) créditos	3
Horas de teoría: dos (2) semanales	3
Horas de práctica: cuatro (4) semanales	3
Duración del período: dieciséis (16) semanas	3
Condición:	3
Modalidad: Virtual	3
Requisitos:	3
PROFESORES	3
Profesor coordinador del curso	3
3.2. Profesor(es) instructor(es) del curso	3
INTRODUCCIÓN AL CURSO	3
OBJETIVOS	4
COMPETENCIAS Y CRITERIOS DE DESEMPEÑO	4
RESULTADOS DE APRENDIZAJE	5
TEMAS	5
PLAN DE TRABAJO	6
Metodología	6
Sesiones de teoría	6
Sesiones de práctica (laboratorio o taller)	6
SISTEMA DE EVALUACIÓN	6
REFERENCIAS BIBLIOGRÁFICAS	7

UNIVERSIDAD DE INGENIERÍA Y TECNOLOGÍA

SILABO 2021-1

1. ASIGNATURA

CS2101 - Teoría de la Computación

2. DATOS GENERALES

2.1 Ciclo: 4°

2.2 Créditos: cuatro (4) créditos

2.3 Horas de teoría: dos (2) semanales

2.4 Horas de práctica: cuatro (4) semanales

2.5 Duración del período: dieciséis (16) semanas

2.6 Condición:

- Obligatorio para Ciencia de la Computación

2.7 Modalidad: Virtual

2.8 Requisitos:

- CS1D02 ESTRUCTURAS DISCRETAS II

3. PROFESORES

3.1 Profesor coordinador del curso

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

Horario de atención: Martes 4pm-5pm

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

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

Horario de atención: Martes 4pm-5pm

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. Entender y aplicar técnicas fundamentales para hacer demostraciones formales sobre estructuras discretas, autómatas y lenguajes.

Sesión 2. Conocer los principales modelos usados en teoría de autómatas así como las diferencias que existen entre ellos. Entender las distintas relaciones que existen entre los modelos de autómatas presentados, así como su relación con los lenguajes regulares.

Sesión 3. Entender todas las formas de representación de los lenguajes regulares. Entender las limitaciones de los lenguajes regulares. Demostrar la existencia de lenguajes no regulares usando el lema de bombeo.

Sesión 4. Entender el concepto de lenguaje independiente del contexto y su relación con los lenguajes regulares. Conocer las principales maneras de representar lenguajes independientes del contexto. Comprender las limitaciones de los lenguajes independientes del contexto y la existencia de lenguajes que no lo son.

Sesión 5. Entender el concepto de algoritmo encapsulado en la definición de Máquina de Turing. Conocer las limitaciones de los algoritmos para resolver problemas computacionales. Entender el concepto de reducción entre problemas computacionales.

Sesión 6. Conocer las limitaciones en tiempo y memoria de los algoritmos para resolver problemas resolubles. Comprender el concepto de problema NP-completo y su importancia en la teoría de la computación. Aplicar reducciones entre problemas para demostrar su dificultad.

6. COMPETENCIAS Y CRITERIOS DE DESEMPEÑO

Los criterios de desempeño que se van a trabajar en este curso son:

2.1: Identifica problemas complejos de computación y otras disciplinas relevantes en el dominio (nivel 2).

2.3: Investiga literatura de computación y otras disciplinas relevantes en el dominio (nivel 2).

3.2. Diseña, implementa y evalúa soluciones a problemas complejos de computación (nivel 2).

9.1. Reconoce la necesidad del aprendizaje autónomo (nivel 2).

7. RESULTADOS DE APRENDIZAJE

Al final del curso de Teoría de la Computación se espera que el estudiante sea capaz de:

RA1. Encontrar técnicas similares a las vistas en clase para analizar problemas de teoría de autómatas.

RA2. Examinar la literatura más relevante en los tópicos de autómatas vistos en clase.

RA3. Implementar software eficiente para resolver problemas en autómatas

RA4. Hacer uso de la guía de ejercicios para complementar el aprendizaje en los tópicos más relevantes de teoría de la computación

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

EVALUACIÓN	TEORÍA	PRÁCTICA Y/O LABORATORIO
-------------------	---------------	-------------------------------------

	Examen E1 (20%) Examen E2 (20%) Examen E3 (20%) Examen E4 (20%)	Proyecto P1 (20%)
	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. SESIONES DE APOYO O TUTORÍAS

Este apartado permite formalizar los espacios de apoyo a los estudiantes y que éstos tengan la atención NECESARIA y el tiempo disponible para presentar sus dudas y consultas acerca del curso:

Semana	Fecha/ Hora	Tema a tratar	Objetivos de la sesión
2	27/04/2021 4:00 PM – 5:00 PM	Evaluaciones	Resolver dudas sobre sistema de evaluación y temas a tratar en el curso
4	11/05/2021 4:00 PM – 5:00 PM	Tópicos del curso	Resolver dudas sobre primeras 3 semanas
6	25/05/2021 4:00 PM – 5:00 PM	Tópicos del curso	Resolver dudas sobre primeras 5 semanas
8	08/06/2021 4:00 PM – 5:00 PM	Parcial	Reclamos examen parcial
10	22/06/2021 4:00 PM – 5:00 PM	Tópicos del curso	Resolver dudas sobre temas tratados en clase
12	06/07/2021 4:00 PM – 5:00 PM	Proyecto	Resolver dudas sobre entrega del proyecto

Fecha de actualización: 09/04/2021

Revisado y aprobado por el Centro de Excelencia en Enseñanza y Aprendizaje y la Dirección de Ciencia de la Computación

14	20/07/2021 4:00 PM – 5:00 PM	Final	Consultas sobre ultimas notas y examen final
----	---------------------------------	-------	--

12. 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