



I. INFORMACIÓN GENERAL

CURSO	:	Algoritmos y Estructuras de Datos
CÓDIGO	:	CC182
CICLO	:	202401
CUERPO ACADÉMICO	:	Arias Orihuela, John Edward Diaz Suarez, Jorge Eduardo Montoya Cubas, Carlos Fernando SALAS ARBAIZA, CESAR ENRIQUE Sopla Maslucán, Abraham Zubieta Cárdenas, Robert Ernesto
CRÉDITOS	:	4
SEMANAS	:	16
HORAS	:	4 H (Teoría) Semanal
ÁREA O CARRERA	:	Ciencias de la Computacion

II. MISIÓN Y VISIÓN DE LA UPC

Misión: Formar líderes íntegros e innovadores con visión global para que transformen el Perú.

Visión: Ser líder en la educación superior por su excelencia académica y su capacidad de innovación.

III. INTRODUCCIÓN

El curso de Algoritmos y Estructuras de Datos realiza un análisis de las aplicaciones de mayor uso y del proceso de construcción de estas.

Explora el desarrollo de sistemas de software de alta complejidad para el manejo de un gran volumen de datos y presenta fundamentos sólidos de técnicas de programación, haciendo énfasis en el análisis y diseño de tipos de datos abstractos para diversos propósitos.

Técnicas sistemáticas con base en matemáticas para analizar la complejidad asintótica de algoritmos, es la primera línea de herramientas necesarias para profesionales en Ciencias de Computación e Ingeniería de Software, para el aseguramiento de la calidad de un producto software y al mismo tiempo permiten considerar el impacto ambiental al ofrecernos una ventana a aspectos del uso de energía por las unidades de procesamiento y su relación con la complejidad de los algoritmos. E

El curso busca que los estudiantes experimenten con diversas herramientas en el desarrollo de videojuegos los cuales son usados como una fuente de problemas de alta complejidad y un medio de motivación.

El curso de Algoritmos y Estructuras de Datos se dicta en el III ciclo para las carreras de Ciencias de Computación e Ingeniería de Software.

El curso ha sido diseñado con el propósito de permitir al futuro profesional, desarrollar su capacidad para construir aplicaciones que hacen uso de grandes cantidades de datos, potenciar disciplinas como: inteligencia artificial, computación gráfica, algoritmos entre otras, así como de problemas de alta complejidad.

IV. LOGRO (S) DEL CURSO

Al finalizar el curso, el estudiante construye aplicaciones para el manejo de grandes cantidades de datos de manera ingeniosa.

Competencia general: Pensamiento Innovador

Nivel de logro: 1

Competencia específica: ABET 2 - Diseño de solución

Nivel de logro: 1

V. UNIDADES DE APRENDIZAJE

UNIDAD N°: 1 Colecciones base y ordenamiento
<p>LOGRO</p> <p>Competencia(s):</p> <ul style="list-style-type: none">- Diseño de solución- Pensamiento Innovador <p>Logro: Al finalizar la unidad, el estudiante implementa una solución basada en principios de complejidad de tiempo y espacio computacional creativamente.</p> <p>TEMARIO</p> <p>Contenido 1:</p> <ul style="list-style-type: none">- Análisis de algoritmos.- Generalización- Archivos <p>Actividades de aprendizaje:</p> <ul style="list-style-type: none">- Exposición participativa- Resolución de ejercicios <p>Bibliografía:</p> <p>Cormen, H., et. al. (2009) Introduction to Algorithms, MIT Press. Cap 2, 3.</p> <p>Weiss M. A., (2014) Data structures and algorithm analysis in C++, Pearson, Cap 1.</p> <p>Contenido 2:</p> <ul style="list-style-type: none">- Recursividad- Lambdas <p>Actividades de aprendizaje:</p> <ul style="list-style-type: none">- Aprendizaje basado en problemas- Resolución de ejercicios <p>Evidencias de aprendizaje: PC1</p> <p>Bibliografía:</p> <p>(2009) Introduction to Algorithms, MIT Press. Cap 4.</p> <p>Weiss M. A., (2014)</p> <p>Data structures and algorithm analysis in C++, Pearson, Cap 1.</p> <p>Contenido 3:</p> <ul style="list-style-type: none">- Listas enlazadas <p>Actividades de aprendizaje:</p> <ul style="list-style-type: none">- Aprendizaje basado en problemas- Resolución de ejercicios

Bibliografía:

Cormen, H., et. al. (2009) Introduction to Algorithms, MIT Press. Cap 10.

Weiss M. A., (2014) Data structures and algorithm analysis in C++, Pearson, Cap 3.

Contenido 4:

- Casos especiales de listas enlazadas

Actividades de aprendizaje:

- Aprendizaje basado en problemas

- Resolución de Ejercicios

Evidencias de aprendizaje: TP

Construye estructuras de datos en grupos pequeños a través del Desarrollo un proyecto aplicado.

Bibliografía:

Cormen, H., et. al. (2009) Introduction to Algorithms, MIT Press. Cap 10.

Contenido 5:

- Pilas y colas

Actividades de aprendizaje:

- Aprendizaje basado en problemas

- Resolución de ejercicios

Evidencias de aprendizaje: TP

Construye estructuras de datos en grupos pequeños a través del Desarrollo un proyecto aplicado.

Bibliografía:

Cormen, H., et. al. (2009) Introduction to Algorithms, MIT Press. Cap 10.

Contenido 6:

- Fisher-Yates shuffle

- Algoritmos de ordenamiento básicos

Actividades de aprendizaje:

- Aprendizaje basado en problema

- Resolución de ejercicios

Evidencias de aprendizaje: TP

Bibliografía:

Cormen, H., et. al. (2009) Introduction to Algorithms, MIT Press. Cap 2.

Weiss M. A., (2014) Data structures and algorithm analysis in C++, Pearson, Cap 7.

Contenido 7:

- Algoritmos de ordenamiento (selección, inserción, Shell)

Actividades de aprendizaje:

- Actividad grupal

- Resolución de ejercicios

Evidencias de aprendizaje: TP

Construye estructuras de datos en grupos pequeños a través del Desarrollo un proyecto aplicado.

Bibliografía:

Cormen, H., et. al. (2009) Introduction to Algorithms, MIT Press. Cap 6, 7, 8.

Contenido 8: Examen Parcial (EA)

HORA(S) / SEMANA(S)

Semanas 1, 2, 3, 4, 5, 6, 7, 8

UNIDAD N°: 2 Búsqueda y árboles

LOGRO

Competencia(s):

- Diseño de solución
- Pensamiento Innovador

Logro: Al finalizar la unidad, el estudiante construye aplicaciones basadas en el procesamiento de grandes cantidades de datos considerando el tiempo de respuesta creativamente.

TEMARIO

Contenido 9: Algoritmos de ordenamiento avanzado (merge, quick)

Actividades de aprendizaje:

Aprendizaje basado en problema

Resolución de ejercicios

Evidencias de aprendizaje: PC2

Bibliografía:

Cormen, H., et. al. (2009) Introduction to Algorithms, MIT Press. Cap 6, 11. Weiss M. A., (2014) Data structures and algorithm analysis in C++, Pearson, Cap 5.

Web: Git Handbook- Github Guides En: <https://guides.github.com/introduction/git-handbook/>

Contenido 10:

- Heaps
- Hash tables

Actividades de aprendizaje:

- Aprendizaje basado en problema

- Resolución de Ejercicios

Evidencias de aprendizaje: PC2

Bibliografía:

Cormen, H., et. al. (2009) Introduction to Algorithms, MIT Press. Cap 12.

Weiss M. A., (2014) Data structures and algorithm analysis in C++, Pearson, Cap 4.

Web: Larn Git Branching En: <https://learngitbranching.js.org/>

Contenido 11:

- Búsqueda binaria
- Árboles binarios

Actividades de aprendizaje:

- Aprendizaje basado en problema

- Resolución de ejercicios

Evidencias de aprendizaje: PC2

Bibliografía:

Cormen, H., et. al. (2009) Introduction to Algorithms, MIT Press. Cap 12.

Contenido 12: Árboles binarios de búsqueda

Actividades de aprendizaje:

- Aprendizaje basado en problema
- Resolución de Ejercicios

Evidencias de aprendizaje: PC2

Aplica búsqueda a la solución de problemas, individualmente a través desarrollo de ejercicios de programación.

Bibliografía:

Cormen, H., et. al. (2009) Introduction to Algorithms, MIT Press. Cap 13.

Contenido 13: Árboles binarios de búsqueda balanceados AVL

Actividades de aprendizaje:

- Aprendizaje basado en problema
- Resolución de ejercicios

Bibliografía:

Cormen, H., et. al. (2009) Introduction to Algorithms, MIT Press. Cap 22.

Weiss M. A., (2014) Data structures and algorithm analysis in C++, Pearson, Cap 9.

Contenido 14: Grafos

Actividades de aprendizaje:

- Aprendizaje basado en problema
- Resolución de ejercicios

Bibliografía:

Weiss M. A., (2014) Data structures and algorithm analysis in C++, Pearson, Cap 12.

Contenido 15: Trabajo final

Evidencias de aprendizaje: TF

Construye un solución completa en grupos pequeños mediante el desarrollo de un proyecto de aplicación - Informe y cronograma - Presentación (PPT)

- Código fuente
- Vídeo

PA Explica el proceso del pensamiento innovador en el TF de manera individual mediante un ensayo

HORA(S) / SEMANA(S)

Semanas 9, 10, 11, 12, 13, 14, 15

UNIDAD N°: 3 Outcome ABET 2 - Diseño de solución

LOGRO

Competencia(s):

- Diseño de solución
- Pensamiento Innovador

Logro: Aplica el diseño de ingeniería para producir soluciones que satisfagan necesidades específicas considerando la salud pública, seguridad y bienestar, así como los factores globales, culturales, sociales, ambientales y económicos

TEMARIO

Contenido 15:

- ABET 2 - Diseño de solución: DD
- Pensamiento Innovador: Presentación Trabajo Final (TF)

Contenido 16: Examen Final (EB)

HORA(S) / SEMANA(S)

Semana 15 y 16

VI. METODOLOGÍA

El Modelo Educativo de la UPC asegura una formación integral, que tiene como pilar el desarrollo de competencias, las que se promueven a través de un proceso de enseñanza aprendizaje donde el estudiante cumple un rol activo en su aprendizaje, construyendo a partir de la reflexión crítica, análisis, discusión, evaluación, exposición e interacción con sus pares, y conectándolo con sus experiencias y conocimientos previos. Por ello, cada sesión está diseñada para ofrecer al estudiante diversas maneras de apropiarse y poner en práctica el nuevo conocimiento en contextos reales o simulados, reconociendo la importancia que esto tiene para su éxito profesional.

El curso es de naturaleza presencial con 5 horas a la semana divididos en 3 horas de laboratorio de cómputo y 2 horas en aula teórica. Las actividades a realizar son: análisis y discusión del contenido teórico del curso seguido de desarrollo de ejemplos y luego desarrollo de ejercicios por parte de los estudiantes, desarrollo de tareas académicas prácticas, ensayos escritos y un proyecto final aplicativo. Los estudiantes deberán poner en práctica el diseño de algoritmos y uso de herramientas de verificación así como capacidad de gestión de proyectos de desarrollo de software. El estudiante deberá dedicar al menos cuatro horas a lecturas y desarrollo de actividades complementarias a la semana fuera del horario de clase.

VII. EVALUACIÓN

FÓRMULA

$$10\% (PC1) + 15\% (TP1) + 15\% (EA1) + 10\% (PC2) + 15\% (TF1) + 20\% (EB1) + 15\% (DD1)$$

TIPO DE NOTA	PESO %
PC - PRÁCTICAS PC	10
TP - TRABAJO PARCIAL	15
EA - EVALUACIÓN PARCIAL	15
PC - PRÁCTICAS PC	10
DD - EVAL. DE DESEMPEÑO	15
TF - TRABAJO FINAL	15
EB - EVALUACIÓN FINAL	20

VIII. CRONOGRAMA

TIPO DE PRUEBA	DESCRIPCIÓN NOTA	NÚM. DE PRUEBA	FECHA	OBSERVACIÓN	RECUPERABLE
PC	PRÁCTICAS PC	1	Semana 4	Evidencia de aprendizaje: Solución entregada Tipo de evaluación: Individual	SÍ
TP	TRABAJO PARCIAL	1	Semana 7	Evidencia de aprendizaje: Solución entregada Tipo de evaluación: Grupal	NO
EA	EVALUACIÓN PARCIAL	1	Semana 8	Evidencia de aprendizaje: Solución entregada Tipo de evaluación: Individual	SÍ
PC	PRÁCTICAS PC	2	Semana 12	Evidencia de aprendizaje: Solución entregada Tipo de evaluación: Individual	SÍ
DD	EVAL. DE DESEMPEÑO	1	Semana 15	Evidencia de aprendizaje: Solución entregada Tipo de evaluación: Individual	NO
TF	TRABAJO FINAL	1	Semana 15	Evidencia de aprendizaje: Solución entregada Tipo de evaluación: Grupal	NO
EB	EVALUACIÓN FINAL	1	Semana 16	Evidencia de aprendizaje: Solución entregada Tipo de evaluación: Individual	SÍ

IX. BIBLIOGRAFÍA DEL CURSO

https://upc.alma.exlibrisgroup.com/leganto/readinglist/lists/10440296500003391?institute=51UPC_INST&auth=LOCAL

X. RED DE APRENDIZAJE

0 - copia - copia.jpg

ANEXO

En este anexo, se encuentran los reglamentos que todo alumno está obligado a leer y a cumplir en su rol de estudiante universitario en la UPC.

REGLAMENTO DE DISCIPLINA DE ALUMNOS :

<https://sica.upc.edu.pe/categoria/reglamentos-upc/sica-reg-26-reglamento-de-disciplina-de-alumnos>

REGLAMENTO PARA LA PREVENCIÓN E INTERVENCIÓN EN CASOS DE HOSTIGAMIENTO SEXUAL- UPC:

<https://sica.upc.edu.pe/categoria/normalizacion/sica-reg-31-reglamento-para-la-prevencion-e-intervencion-en-casos-de-hostiga>