

Programa de curso
Estructuras de Datos

Carrera: Ingeniería de Sistemas de Información con grado en Bachillerato y salida lateral de Diplomado en Programación de Aplicaciones Informáticas

Código de Carrera: ISIN (ingeniería Sistemas Información)

Nombre del curso:	Estructuras de Datos																																							
Tipo de curso:	Regular																																							
Código de curso:	EIF-207																																							
Nivel y grado académico:	3 ^{er} nivel Grado: Bachillerato y salida lateral de Diplomado en Programación de Aplicaciones Informáticas																																							
Periodo lectivo:	2 ^{do} ciclo 2023																																							
Modalidad:	Presencial																																							
Naturaleza:	Teórico/práctico																																							
Créditos:	4																																							
Horas totales semanales:	11																																							
Horas del curso:	Teoría: 2-4 Laboratorio: 2-4 Estudio independiente: 7																																							
Horas docentes:	<table><tr><th>Grupo</th><th>NRC</th><th>Profesor</th><th>Días</th><th>Horas</th></tr><tr><td>01</td><td>50219</td><td></td><td>M, V</td><td>8:00-9:40</td></tr><tr><td>02</td><td>50221</td><td>Georges Alfaro S.</td><td>M, V</td><td>10:00-11:40</td></tr><tr><td>03</td><td>50222</td><td>Georges Alfaro S.</td><td>M, V</td><td>13:00-14:40</td></tr><tr><td>04</td><td>50223</td><td>Georges Alfaro S.</td><td>M, V</td><td>15:00-16:40</td></tr><tr><td>05</td><td>50225</td><td>José Pablo Calvo S.</td><td>M, V</td><td>18:00-19:40</td></tr><tr><td>06</td><td>50226</td><td>José Pablo Calvo S.</td><td>M, V</td><td>20:00-21:40</td></tr></table>					Grupo	NRC	Profesor	Días	Horas	01	50219		M, V	8:00-9:40	02	50221	Georges Alfaro S.	M, V	10:00-11:40	03	50222	Georges Alfaro S.	M, V	13:00-14:40	04	50223	Georges Alfaro S.	M, V	15:00-16:40	05	50225	José Pablo Calvo S.	M, V	18:00-19:40	06	50226	José Pablo Calvo S.	M, V	20:00-21:40
Grupo	NRC	Profesor	Días	Horas																																				
01	50219		M, V	8:00-9:40																																				
02	50221	Georges Alfaro S.	M, V	10:00-11:40																																				
03	50222	Georges Alfaro S.	M, V	13:00-14:40																																				
04	50223	Georges Alfaro S.	M, V	15:00-16:40																																				
05	50225	José Pablo Calvo S.	M, V	18:00-19:40																																				
06	50226	José Pablo Calvo S.	M, V	20:00-21:40																																				
Horario de atención a estudiantes:	<table><tr><th>Profesor</th><th>Horario</th></tr><tr><td>José Pablo Calvo Suárez</td><td>Martes y viernes de 5:00pm a 6:00pm</td></tr><tr><td>Georges Alfaro Salazar (coordinador)</td><td>Miércoles de 2:00pm a 6:00pm</td></tr></table>					Profesor	Horario	José Pablo Calvo Suárez	Martes y viernes de 5:00pm a 6:00pm	Georges Alfaro Salazar (coordinador)	Miércoles de 2:00pm a 6:00pm																													
Profesor	Horario																																							
José Pablo Calvo Suárez	Martes y viernes de 5:00pm a 6:00pm																																							
Georges Alfaro Salazar (coordinador)	Miércoles de 2:00pm a 6:00pm																																							
Horario del curso:	(el mismo que corresponde a las horas docentes indicadas para el grupo respectivo)																																							

Requisitos	EIF-203 Estructuras Discretas para Informática EIF-204 Programación 2						
Correquisitos	No existen						
Área disciplinaria	Formación complementaria						
Docentes;	<table> <tr> <th>Profesor</th><th>Correo electrónico</th></tr> <tr> <td>José Pablo Calvo Suárez</td><td>jose.calvo.suarez@una.ac.cr</td></tr> <tr> <td>Georges Alfaro Salazar</td><td>georges.alfaro@gmail.com georges.alfaro.salazar@una.cr</td></tr> </table>	Profesor	Correo electrónico	José Pablo Calvo Suárez	jose.calvo.suarez@una.ac.cr	Georges Alfaro Salazar	georges.alfaro@gmail.com georges.alfaro.salazar@una.cr
Profesor	Correo electrónico						
José Pablo Calvo Suárez	jose.calvo.suarez@una.ac.cr						
Georges Alfaro Salazar	georges.alfaro@gmail.com georges.alfaro.salazar@una.cr						

En esta Universidad nos comprometemos a prevenir, investigar y sancionar el hostigamiento sexual entendido como toda conducta o comportamiento físico, verbal, no verbal, escrito, expreso o implícito, de connotación sexual, no deseado o molesto para quien o quienes lo reciben, reiterado o aislado. Si usted está siendo víctima de hostigamiento diríjase a la Fiscalía de Hostigamiento Sexual de la UNA o llame al teléfono: 2277-3961.

Descripción

El curso presenta conceptos básicos de estructuras de datos y de análisis y construcción de algoritmos. Se hace énfasis en buenas prácticas para la construcción, validación y verificación del software. La práctica consiste en el desarrollo de algoritmos eficaces y eficientes para resolver problemas computacionales típicos. Se profundizará sobre conceptos de estructuras de datos y algoritmos, para lo cual se supone que el estudiante ya posee conocimientos básicos sobre el tema y conoce las técnicas fundamentales de programación orientada a objetos.

Objetivos

Objetivo general

Conocer los fundamentos de la conceptualización e implementación de estructuras de datos básicas y de los algoritmos relacionados. También se busca conocer las principales aplicaciones de dichas estructuras de datos.

Objetivos específicos

1. Conocer las principales estructuras de datos y su aplicación adecuada, según las características de los problemas a resolver.
2. Administrar eficaz y eficientemente el espacio y tiempo computacionales.
3. Identificar los conceptos relacionados con el análisis de algoritmos.
4. Clasificar adecuadamente cada una de las diferentes estructuras de datos y sus principales características.
5. Conocer los principales algoritmos utilizados para la solución de problemas representativos.
6. Hacer un uso adecuado de la memoria dinámica para la representación de estructuras de datos.

Habilidades y destrezas

A través del curso se pretende facilitar y promover el desarrollo de al menos las siguientes competencias del estudiante:

- Capacidad para el desarrollo de algoritmos recursivos e iterativos
- Capacidad para relacionar los aspectos teóricos de la resolución de problemas y representación de datos con los aspectos prácticos del almacenamiento y manipulación de información.
- Habilidad para trabajar en equipo y resolver conflictos de manera efectiva.
- Poder dividir tareas efectiva y adecuadamente.
- Colaborar para la obtención de un fin.

Contenidos

1. Introducción

- Tipos de datos abstractos
- Análisis de algoritmos y notación asintótica
- Recursividad general

2. Estructuras lineales

- Listas
- Pilas (*stacks*)
- Colas (*queues*)
- Implementación de estructuras lineales
 - Almacenamiento contiguo
 - Almacenamiento enlazado
 - Listas simples y dobles
 - Colas con prioridad
- Bibliotecas especializadas para estructuras de datos (p.ej. STL – C++)

3. Algoritmos de ordenamiento

- Análisis de los algoritmos de ordenamiento
- Ordenamiento simple
 - Algoritmo de ordenamiento por burbuja (*bubble sort*)
 - Algoritmo de ordenamiento por inserción (*insertion sort*)
 - Algoritmo de ordenamiento por selección (*selection sort*)
- Otros algoritmos de ordenamiento
 - Ordenamiento por combinación (*merge sort*)
 - Ordenamiento rápido (*quicksort*)
 - Ordenamiento por montículos (*heap sort*)

4. Introducción a las estructuras de datos no lineales

- Árboles
- Estructuras asociativas
- Grafos

5. Árboles

- Árboles generales
- Árboles binarios
- Árboles balanceados
- Árboles binarios de búsqueda

6. Estructuras asociativas y algoritmos de búsqueda

- Tablas de dispersión (*hash tables*)
- Diccionarios
- Algoritmos de búsqueda
 - Búsqueda binaria
 - Búsqueda asociativa

7. Grafos

- Formas de representación
 - Matrices de adyacencia
 - Representación lineal (listas)
 - Representación no lineal (árboles)
- Algoritmos fundamentales
 - Búsqueda en profundidad (*depth first*)
 - Búsqueda en anchura (*breadth first*)
 - Determinar caminos más cortos
 - Árbol de expansión de costo mínimo

8. Estructuras y algoritmos avanzados

- Programación dinámica
- Algoritmos ávidos (*greedy algorithms*)
- Algoritmo de búsqueda por retroceso (*backtracking*)
- Árboles B, B+, B*

Metodología

El curso contará con la exposición magistral por parte del profesor de cada uno de los contenidos descritos. Además, se harán sesiones de resolución de problemas y prácticas de laboratorio para enfrentar al estudiante de manera directa con las principales dificultades y técnicas utilizadas en programación. Así se logra también que el estudiante aplique los conceptos expuestos en situaciones específicas.

De la misma manera, habrá proyectos de programación que requieren más tiempo del que se dispone durante las lecciones, donde los estudiantes resuelven en grupo ejercicios de dificultad media o alta, para conocer, estudiar y resolver problemas representativos.

Se promoverá la investigación constructiva en los estudiantes, de tal manera que estos brinden aportes al aprendizaje propio y de los demás y construyan sus propias soluciones a los problemas planteados.

Evaluación

La evaluación se hará contemplando sus tres funciones: diagnóstica, formativa y sumativa. Esta última se estructurará de la siguiente manera:

Rubro de evaluación	Valor
Primer examen parcial (Comprende los temas 1,2,3,4 y 5)	20%
Examen final (Cubre todos los temas del curso, y en particular, los temas 6, 7 y 8)	20%
Proyectos programados	40%
Actividades que promueven el aprendizaje (Tareas, laboratorios y/o pruebas cortas) El docente puede realizar evaluaciones cortas sin necesidad de avisar previamente su aplicación. También se asignarán tareas cortas y prácticas que pueden ser completadas en clase o asignadas para ser trabajadas fuera de las horas de clase.	20%
Total	100%

Los exámenes en el curso buscan medir y evaluar la comprensión de cada estudiante del material estudiado durante el curso y del trabajo realizado en los proyectos. Los exámenes deben realizarse y entregarse individualmente. Cuando algún trabajo, práctica o proyecto se realice en grupos, la conformación de dichos grupos de trabajo en cada proyecto quedará a criterio de cada uno de los docentes.

Por la naturaleza del curso, es inevitable que los contenidos desarrollados sean acumulativos para los exámenes y trabajos prácticos. Es decir, aunque en una evaluación (sea individual o grupal) se deba profundizar en uno o más temas específicos, esto no implica que no se puedan incluir temas anteriormente evaluados.

Al ser un curso cuya evaluación contempla aspectos prácticos, como los laboratorios y proyectos programados, no hay examen extraordinario. La suma de los porcentajes obtenidos por el estudiante en los rubros anteriores determina su nota de aprovechamiento (NA). El curso se aprueba con una NA igual o superior al 70%.

Especificaciones generales

En caso de corroborarse algún fraude en la aplicación de alguna evaluación escrita o en la documentación, algoritmos o implementación de las tareas o proyectos, la Escuela de Informática aplicará las sanciones establecidas en el reglamento interno de la Universidad Nacional.

Normas específicas para la ejecución del curso

Es requisito indispensable para ganar el curso la presentación de todos los proyectos.

Los exámenes se realizarán de manera coordinada por todos los profesores, preferiblemente un único día para todos los grupos.

Cada profesor utilizará el o los lenguajes de programación que considere pertinente para el desarrollo del curso (C++, Python, Java) y cualquier otra herramienta de *software* adecuada.

Cronograma

El II ciclo lectivo inicia el 31 de julio y finaliza el 18 de noviembre de 2023. El siguiente cronograma es una guía para el profesor y los estudiantes sobre los temas a tratarse durante el curso. Cada profesor definirá con sus estudiantes las fechas de los entregables, trabajos en clase, proyectos, pruebas cortas (*quices*) y otros mecanismos de evaluación. El cronograma puede variar dependiendo del desarrollo de los temas, de los feriados y otras actividades que sean relevantes para el curso.

Se m.	Fecha	Contenidos y aprendizajes integrales	Actividades
1	De 31-07-2023 hasta 05-08-2023	Inicio de lecciones. Tema 1 (Introducción, tipos de datos abstractos, análisis de algoritmos)	Presentación y bienvenida por parte del profesor. Lectura y revisión de la carta al estudiante.
2	De 07-08-2023 hasta 12-08-2023	Tema 1 (Introducción, tipos de datos abstractos, análisis de algoritmos)	
3	De 14-08-2023 hasta 19-08-2023	Tema 2 (Estructuras lineales)	Entrega del enunciado del primer proyecto, del parte del docente
4	De 21-08-2023 hasta 26-08-2023	Tema 2 (Implementación de estructuras lineales)	
5	De 28-08-2023 hasta 02-09-2023	Tema 3 (Algoritmos de ordenamiento)	
6	De 04-09-2023 hasta 09-09-2023	Tema 4 (Estructuras de datos no lineales)	
7	De 11-09-2023 hasta 16-09-2023	Tema 5 (árboles)	
8	De 18-09-2023 hasta 23-09-2023		Entrega de la solución del primer proyecto, de parte de los estudiantes. 1er examen parcial.
9	De 25-09-2023 hasta 30-09-2023	Tema 6 (Estructuras asociativas y algoritmos de búsqueda)	
10	De 02-10-2023 hasta 07-10-2023		Entrega del enunciado del segundo proyecto, de parte del docente
11	De 09-10-2023 hasta 14-10-2023		
12	De 16-10-2023 hasta 21-10-2023	Tema 7 (Grafos)	
13	De 23-10-2023 hasta 28-10-2023		
14	De 30-10-2023 hasta 04-11-2023	Tema 8 (Estructuras y algoritmos avanzados)	
15	De 06-11-2023 hasta 11-11-2023		Entrega de la solución del segundo proyecto, de parte de los estudiantes.
16	De 13-11-2023 hasta 18-11-2023		Examen final

Bibliografía

Se recomienda hacer uso de las bases de datos del SIDUNA (Sistemas e Información Documental de la Universidad Nacional) para acceder a fuentes recomendadas de forma que se fomente el uso de los recursos de la universidad.

Bibliografía base del curso

- **Morin, Pat (2013).** Open Data Structures. Au Press, USA
- **Cormen, Thomas H., Leiserson, Charles E. (2009).** Introduction to Algorithms. MIT Press, USA. 3rd Edition
- **Aho, Alfred V., Hopcroft, John E., Ullman, Jeffrey D. (1988).** Estructuras de Datos y Algoritmos. Addison-Wesley, México.

Bibliografía de referencia.

- **Shaffer, Clifford A. (2012).** *Data Structures and Algorithm Analysis*. Department of Computer Science, Virginia Tech. Dover Publications, USA.
- **Goodrich, Michael T, Tamassia, Roberto, Mount, David M. (2011).** *Data Structures and Algorithms in C++*. John Wiley & Sons, Inc. 2nd Edition.
- **Goodrich, Michael T, Tamassia, Roberto, Goldwasser, Michael H. (2014).** *Data Structures and Algorithms in Java*. John Wiley & Sons, Inc.
- **Goodrich, Michael T, Tamassia, Roberto, Goldwasser, Michael H. (2016).** *Data Structures and Algorithms in Python*. John Wiley & Sons, Inc.
- **Drozdek, Adam (2012).** *Data Structures and Algorithms in C++*. Cengage Learning, 4th Edition.
- **Langsam, Yedidyah, Augenstein, Moshe J., et al. (1995).** *Data Structures Using C and C++*. Prentice Hall. 2nd Edition.

Actividades extracurriculares

- Se promoverá la participación de los estudiantes en actividades extracurriculares organizadas por la cátedra o la Escuela de Informática de la Universidad Nacional, como complemento a los cursos, donde los estudiantes podrán aplicar algunos de los conocimientos adquiridos y fomentar también la investigación y las técnicas de resolución de problemas.
- Se planificará una actividad que tenga relación con temas del área de programación o resolución de problemas, de acuerdo con el interés y disponibilidad de los estudiantes.