

# UNIVERSIDAD NACIONAL DE SAN AGUSTÍN DE AREQUIPA



## VICERRECTORADO ACADÉMICO

FACULTAD DE INGENIERÍA DE PRODUCCIÓN Y SERVICIOS  
DEPARTAMENTO ACADÉMICO DE INGENIERIA DE SISTEMAS E INFORMATICA

SÍLABO 2024 - A

ASIGNATURA: ANALISIS Y DISEÑO DE ALGORITMOS

### 1. INFORMACIÓN ACADÉMICA

<b>Periodo académico:</b>	2024 - A	
<b>Escuela Profesional:</b>	CIENCIA DE LA COMPUTACIÓN	
<b>Código de la asignatura:</b>	1703131	
<b>Nombre de la asignatura:</b>	ANALISIS Y DISEÑO DE ALGORITMOS	
<b>Semestre:</b>	V (quinto)	
<b>Duración:</b>	17 semanas	
<b>Número de horas (Semestral)</b>	<b>Teóricas:</b>	2.00
	<b>Prácticas:</b>	2.00
	<b>Seminarios:</b>	0.00
	<b>Laboratorio:</b>	2.00
	<b>Teórico-prácticas:</b>	0.00
<b>Número de créditos:</b>	4	
<b>Prerrequisitos:</b>	ALGORITMOS Y ESTRUCTURAS DE DATOS (1702224)	

### 2. INFORMACIÓN DEL DOCENTE, INSTRUCTOR, COORDINADOR

DOCENTE	GRADO ACADÉMICO	DPTO. ACADÉMICO	HORAS	HORARIO
PAZ VALDERRAMA, ALFREDO		INGENIERIA DE SISTEMAS E INFORMATICA	8	Mar: 14:00-15:40 Jue: 14:00-15:40
PAZ VALDERRAMA, ALFREDO		INGENIERIA DE SISTEMAS E INFORMATICA	6	Mié: 15:50-17:30 Jue: 15:50-17:30

### 3. INFORMACIÓN ESPECIFICA DEL CURSO (FUNDAMENTACIÓN, JUSTIFICACIÓN)

Los algoritmos resuelven problemas mediante el uso de datos de entrada y siguiendo pasos finitos, ordenados y bien definidos con el fin de alcanzar la salida deseada. El estudio teórico del desempeño de los algoritmos permite evaluar si una solución a un problema es factible usando los recursos computacionales (usualmente tiempo y espacio) disponibles. Donald Knuth, dijo: 'Un algoritmo debe ser

visto como más que una forma de hacer una tarea. Es una excelente oportunidad para aprender a pensar de manera clara y precisa. [...] Los algoritmos son la esencia de la computación'. En este curso se estudiarán el desempeño de las mayores técnicas de diseño de algoritmos y su aplicación en distintos tipos de problemas; también se abordarán los límites de los problemas que pueden ser computables.

#### 4. COMPETENCIAS/OBJETIVOS DE LA ASIGNATURA

Desarrollar la capacidad para evaluar la complejidad y calidad de algoritmos propuestos para un determinado problema.

Estudiar los algoritmos más representativos, introductorios de las clases más importantes de problemas tratados en computación.

Desarrollar la capacidad de resolución de problemas algorítmicos utilizando los principios fundamentales de diseño de algoritmos aprendidos.

Ser capaz de responder a las siguientes preguntas cuando le sea presentado un nuevo algoritmo: ¿Cuán buen desempeño tiene?, ¿Podrá existir una mejor forma de resolver el problema?.

#### 5. CONTENIDO TEMATICO

##### PRIMERA UNIDAD

###### Capítulo I: Introducción

**Tema 01:** Motivación: Importancia de los Algoritmos en la solución de problemas., Reglas del curso

**Tema 02:** Examen de Entrada

###### Capítulo II: Análisis de Algoritmos

**Tema 03:** Análisis asintótico y notación.

**Tema 04:** Comparación de la tendencia de las funciones.

**Tema 05:** Definición formal y demostración de propiedades de la notación asintótica.

**Tema 06:** Análisis del mejor, peor y caso promedio: insertion sort, merge sort.

**Tema 07:** Técnicas de solución de recurrencias: árbol de recursión, inducción matemática, funciones generatrices, el teorema maestro.

**Tema 08:** Incidencia de factores externos y la implementación de algoritmos.

###### Capítulo III: Diseño de Algoritmos

**Tema 09:** Algoritmos de fuerza bruta.

**Tema 10:** Divide y vencerás: Strassen, Fibonacci, Multiplicación polinomial, Quicksort.

**Tema 11:** Algoritmos avaros: Minimum Spanning Trees.

**Tema 12:** Programación dinámica: la subsecuencia común más larga

**Tema 13:** La ruta más corta: Dijkstra, Floyd, Bellman-Ford, programación lineal, Floyd-Warshall, Johnson.

**Tema 14:** Backtracking., Heurísticas.

###### Capítulo IV: Las clases de complejidad P y NP

**Tema 15:** Definición formal de las clases P y NP., NP completitud (Teorema de Cook).

**Tema 16:** Problemas NP-completos y demostraciones

#### 6. ESTRATEGIAS DE ENSEÑANZA APRENDIZAJE

### 6.1. Métodos

Método expositivo en las clases teóricas

Método del trabajo independiente para la solución de problemas prácticos en clase y laboratorio donde apliquen los contenidos de la asignatura

Medios: aula virtual, meet, chat, git, etc.

El método de descubrimiento se utilizará cuando se estudie los temas del curso.

### 6.2. Medios

Material bibliográfico, Aula virtual, Presentaciones, Laptop, Lecturas seleccionadas, Hojas de ejercicios.

### 6.3. Formas de organización

Componente académico: la clase magistral con participación activa de los alumnos.

Componente práctico: las prácticas y laboratorios.

Componente investigativo: trabajo final de investigación formativa.

### 6.4. Programación de actividades de investigación formativa y responsabilidad social

Se formarán equipos de no más de 3 estudiantes para validar artículos respecto a algoritmos y su desempeño, lo que ayudará al desarrollo las habilidades investigativas del estudiante.

Se desarrollarán trabajos de campo que vinculen los conocimientos y habilidades adquiridas en la asignatura o el nivel en el que se encuentran con las necesidades de la comunidad

## 7. CRONOGRAMA ACADÉMICO

SEMANA	TEMA	DOCENTE	%	ACUM.
1	Motivación: Importancia de los Algoritmos en la solución de problemas., Reglas del curso	A. Paz	6	6.00
2	Examen de Entrada	A. Paz	6	12.00
3	Análisis asintótico y notación.	A. Paz	6	18.00
4	Comparación de la tendencia de las funciones.	A. Paz	7	25.00
5	Definición formal y demostración de propiedades de la notación asintótica.	A. Paz	6	31.00
6	Análisis del mejor, peor y caso promedio: insertion sort, merge sort.	A. Paz	6	37.00
7	Técnicas de solución de recurrencias: árbol de recursión, inducción matemática, funciones generatrices, el teorema maestro.	A. Paz	6	43.00
8	Incidencia de factores externos y la implementación de algoritmos.	A. Paz	7	50.00
9	Algoritmos de fuerza bruta.	A. Paz	6	56.00
10	Divide y vencerás: Strassen, Fibonacci, Multiplicación polinomial, Quicksort.	A. Paz	6	62.00
11	Algoritmos avaros: Minimum Spanning Trees.	A. Paz	6	68.00
12	Programación dinámica: la subsecuencia común más larga	A. Paz	7	75.00
13	La ruta más corta: Dijkstra, Floyd, Bellman-Ford, programación lineal, Floyd-Warshall, Johnson.	A. Paz	6	81.00
14	Backtracking., Heurísticas.	A. Paz	6	87.00
15	Definición formal de las clases P y NP., NP completitud (Teorema de Cook).	A. Paz	6	93.00
16	Problemas NP-completos y demostraciones	A. Paz	7	100.00

## 8. ESTRATEGIAS DE EVALUACIÓN

### 8.1. Evaluación del aprendizaje

Evaluación Escrita: Tres exámenes.

Evaluación Continua: Tareas, intervenciones en clase y proyecto final.

### 8.2. Cronograma de evaluación

EVALUACIÓN	FECHA DE EVALUACIÓN	EXAMEN TEORÍA	EVAL. CONTINUA	TOTAL (%)
Primera Evaluación Parcial	13-05-2024	15%	15%	30%
Segunda Evaluación Parcial	24-06-2024	15%	15%	30%
Tercera Evaluación Parcial	29-07-2024	24%	16%	40%
TOTAL				100%

## 9. REQUISITOS DE APROBACIÓN DE LA ASIGNATURA

Los exámenes se tomarán con apuntes.

Los alumnos tendrán la oportunidad de rezagar un examen parcial con un plazo de 72 horas y por causas debidamente justificadas y autorizadas por la dirección de la escuela.

Posterior a la aplicación de una prueba se realizan las siguientes actividades: solución de las preguntas del examen; acceso de la prueba por parte del estudiante; recalificación cuando es pertinente; publicación de los resultados, después de todas estas actividades la nota es inmodificable.

Los alumnos que no asistan a la solución del examen y no suban la evidencia de haber revisado su nota en clase con una imagen del examen, su nota y firma, tendrán nota cero en el sistema académico.

Las calificaciones se registran en el sistema de la universidad según cronograma.

Los exámenes tienen tres momentos: la rendición del examen, la solución-entrega de notas y la subida del examen firmado-calificado; cada estudiante debe cumplir con estas actividades.

La honestidad será un factor determinante en la evaluación: Los alumnos que tengan actitudes deshonestas en alguna de sus tareas, trabajos o examen tendrán nota 0.

Acto deshonesto: Copiar la solución de otro durante el examen. Esto incluye mirar al compañero o usar medio electrónicos (celular, etc.)

Acto deshonesto: Compartir código fuente: Copiar, cambiar de nombre a las variables, mostrar el código a un compañero, descargar el código de Internet, explicar el código a un compañero. Tener cuidado de no dejar copias de las tareas en lugares públicos.

Acto deshonesto: Consultoría. Recibir ayuda en la solución de la tarea, esta puede ser en persona, por un compañero de años superiores, por foros de discusión en Internet, etc.

Acto deshonesto: Realizar los trabajos individuales en grupo. Las tareas pueden tener soluciones diversas, si estas son individuales no deben reunirse para hacerlas.

Acto deshonesto: Realizar las tareas grupales de manera individual. Que sólo un compañero haga toda la tarea del grupo, que cada integrante del grupo haga una parte de la tarea, pero que no tenga idea de las demás partes. Las tareas en grupo deben ser hechas en grupo, por lo que se requiere coordinación, no sólo en la distribución del trabajo, sino en la solución de los problemas que se puedan presentar. El grupo debe trabajar como un equipo.

Acto honesto: Explicar lo que se pide en la tarea. Se puede pedir ayuda al profesor o los compañeros para entender lo que se pide en la tarea, pero siendo cuidadosos de no explicar la solución, sólo el enunciado de lo que se pide.

Acto honesto: Explicar los temas o conceptos. Si algún tema o concepto no se entiende, fuera del horario

de clase, se puede pedir al profesor o algún compañero ayuda.

Acto honesto: Llevar apuntes. Se pueden llevar apuntes a los exámenes y a las evaluaciones en los laboratorios, estos apuntes podrán ayudar a recordar comandos, códigos, etc.

## **10. BIBLIOGRAFIA: AUTOR, TÍTULO, AÑO, EDITORIAL**

### **10.1. Bibliografía básica obligatoria**

Thomas H Cormen, Charles E Leiserson, Ronald L Rivest, and Clifford Stein. Introduction to algorithms. MIT press, 2022.

Ferri and Vidal. Case-studies on average-case analysis for an elementary course on algorithms. IEEE Transactions on Education, 1999.

Knuth, Donald E. The Art of Computer Programming: Fundamental Algorithms, volume 1-3. Addison-Wesley Professional, 1997.

### **10.2. Bibliografía de consulta**

R. L. Graham, D. E. Knuth, and O. Patashnik. Concrete Mathematics. Addison-Wesley, Reading, USA, 2 edition, 1995. ISBN 0-201-55802-5.

Ian Holyer. The np-completeness of edge-coloring. SIAM Journal on computing, 10(4):718-720, 1981.

Donald E Knuth. Selected papers on computer science. Cambridge University Press, 1996.

G.L. McDowell. Cracking the Coding Interview: 189 Programming Questions and Solutions. CareerCup, 2015. ISBN 9780984782857. URL: <https://books.google.com.pe/books?id=jD8iswEACAAJ>.

Arequipa, 11 de Abril del 2024

**PAZ VALDERRAMA, ALFREDO**