

13205

# Análisis de Algoritmos y Estructuras de Datos



## Presentación de la Asignatura

Departamento de Ingeniería Informática - Semestre 02/2023



# Contenido

- Organización general de cátedra
- Organización de laboratorio
- Evaluaciones de cátedra y laboratorio
- Recomendaciones para un final feliz



# Objetivos

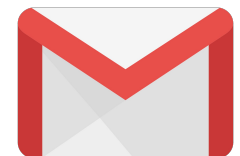
- **Presentación general del curso**
- **Marco de trabajo durante el semestre**
  - Formas de comunicación
  - Objetivos a lograr
  - Contenidos
  - Ponderación de evaluaciones
  - Fechas de evaluaciones
- **Recomendaciones de estudiantes 01/2023**



# Inscripción

Coordinación Docente solicita:

- Mantener sus datos actualizados en LOA (especialmente el correo @usach).
- Inscribirse en uvirtual en el curso ANÁLISIS DE ALGORITMOS Y ESTRUCTURA DE DATOS (13205) con la contraseña entregada por correo usach.
- Mantenerse en contacto a través de los canales oficiales de comunicación (correo @usach.cl, uvirtual).





# Equipo de Trabajo

- PROFESORA:

- Violeta Chang (violeta.chang@usach.cl)  
M2; J2; V2(Lab)

- AYUDANTE:

- Daniel Jara (daniel.jara.p@usach.cl)  
V2(Lab)



# Objetivos del curso

**Propósito:** Resolver problemas usando algoritmos y estructuras de datos adecuados a la situación.

Se espera que estudiantes aprendan a:

- Aplicar los conceptos de tiempo de ejecución y complejidad algorítmica
- Diseñar, implementar y modificar estructuras de datos de acuerdo a las necesidades del problema
- Diseñar algoritmos elementales, iterativos y recursivos
- Recomendar de manera fundada una estructura de datos para un problema dado
- **En laboratorio:** implementar diversas estructuras de datos para problemas específicos.





# Contenidos

Unidad	Materia
Complejidad de algoritmos	Introducción a algoritmos, Pseudocódigo Complejidad algorítmica
	Análisis de complejidad Algoritmos iterativos y recursivos
	Algoritmos de ordenamiento (inserción, selección, burbuja, quicksort)
Estructuras de datos básicas	Conceptos de TDA TDA Lista
	Lista simple, doble, circular
	TDA Pila TDA Cola



# Contenidos

Unidad	Materia
Grafos y sus algoritmos	Grafos: definiciones y tipos de grafos
	Modelamiento de problemas con grafos
	Algoritmos sobre grafos: recorridos, camino mínimo
	Grafos de proximidad MST
Estructuras de datos avanzadas y sus algoritmos	Grafos de proximidad kNN
	Algoritmos sobre grafos: flujo en redes
	TDA Árbol
	Árboles generales, binarios y de búsqueda
	Árboles AVL
	Tablas HASH





# Evaluación

## Cátedra (50%)

Pruebas escritas parciales	PEP1, PEP2, PEP3	33.3% c/u
Cat = promedio(PEP1, PEP2, PEP3)		

## Laboratorio (50%)

Tareas computacionales	L1, L2, L3	25% c/u
Trabajo en sesiones de laboratorio	TL	25%
Lab = promedio(L1, L2, L3, TL)		



# Evaluación

- El promedio final del curso se calcula como:  **$0.5 * \text{Cat} + 0.5 * \text{Lab}$**
- Para aprobar el curso se debe tener 4.0 o más en el promedio de cátedra y 4.0 o más en el promedio de laboratorio, de lo contrario la nota será el mínimo de ellos.
- Cátedra y laboratorio se aprueban por separado.
- La **exigencia académica en evaluaciones de cátedra y laboratorio es de 60%**.
- En caso de inasistencia a una evaluación de cátedra, se debe justificar en secretaría docente dentro de los 5 días hábiles siguientes a la fecha de la evaluación.
- Se considera una prueba optativa de reemplazo (POR) que sustituye la peor nota de cátedra del semestre, que se rinde en la última semana. La **condición para rendir la POR es tener a lo menos 3.6 como promedio de cátedra**.
- Si una tarea se entrega fuera de plazo, se resta un punto por cada hora de retraso.
- Una tarea no entregada tiene calificación 1.0.
- Se evaluará el trabajo individual en las sesiones de laboratorio según corresponda por cronograma. Dichas entregas serán realizadas al término de cada sesión. La nota correspondiente a trabajo en sesiones de laboratorio (TL) corresponde al promedio del 70% mejores notas semanales.



# Evaluación

Fechas	Evaluaciones de cátedra	Evaluaciones de laboratorio
W 20/09		Enunciado Tarea 1
M 03/10	PEP1	
W 04/10		Tarea 1
W 25/10		Enunciado Tarea 2
M 14/11	PEP2	
W 15/11		Tarea 2
W 29/11		Enunciado Tarea 3
M 12/12	PEP3	
W 13/12		Tarea 3
M 19/12	POR	



# Bibliografía / Recursos

- Bibliografía:
  - Estructura de Datos y Algoritmos (Aho, Ullman, Hopcroft)
  - Introduction to Algorithms (Cormen, Leiserson, Rivest, Stein)
- Material disponible en [uvirtual.usach.cl](http://uvirtual.usach.cl)



DEPARTAMENTO DE  
**INGENIERÍA  
INFORMÁTICA**  
UNIVERSIDAD DE SANTIAGO DE CHILE

# Recursos: Comunicación e Información



uVirtual



correo usach



discord



# Recursos: Comunicación e Información

<https://uvirtual.usach.cl>

- Programa (General)
- Cronograma (General)
- Presentación de la asignatura (General)
- Novedades (General)





# Recursos: Comunicación e Información

General	Unidad I	Unidad II	Unidad III	Unidad IV	Tareas de laboratorio				
Inicio	S1-Cat	S1-Ref	S1-Lab	S2-Cat	S2-Ref	S2-Lab	S3-Cat	S3-Ref	S3-Lab

## Unidad 1

- **Título:**
  - Complejidad de algoritmos
- **Duración:**
  - Del 14 de marzo al 08 de abril
- **Contenidos:**
  - Metodología de trabajo y evaluación
  - Introducción a algoritmos, pseudocódigo y complejidad algorítmica
  - Análisis de complejidad, algoritmos iterativos y recursivos
  - Algoritmos de ordenamiento: Inserción, selección, burbuja, quicksort
- **Evaluaciones:**
  - Cátedra:
    - Trabajo en clase - miércoles 23/03 y 30/03
    - Simulacro EC1 - miércoles 06/04
    - EC1 - miércoles 13/04
  - Laboratorio:
    - Trabajo en clase - viernes 25/03, 01/04 y 08/04
    - Tarea 1 - jueves 14/04



# Actividad Semanal (\*)

- Clase de cátedra: martes



Exposición dialogada  
Participación dinámica de estudiantes

- Clase de refuerzo: jueves



Refuerzo y ejercicio: Teoría y práctica  
Trabajo individual y grupal  
Incentivos y simulacros de evaluación

- Clase de laboratorio: viernes



Actividades de implementación en lenguaje C  
Trabajo individual y grupal  
Trabajo en clase con entrega obligatoria

(\*) Por las características del semestre, algunas clases intercambiarán de día para mantener la actividad semanal



# Actividad Semanal: Detalle de Clases

Inicio

S1-Cat

S1-Ref

S1-Lab

S2-Cat

S2-Lab

S3-Cat

S3-Ref

S3-Lab

S4-Cat

S4-Ref

S4-Lab

## Semana 1



**Cátedra**



**Descripción**



**Contenidos**



**Resultados de aprendizaje**



**Actividades antes de la clase**



**Actividades durante la clase**



**Actividades después de la clase**



**Para seguir aprendiendo**



# Dedicación a la asignatura



## Cátedra

- Realizar actividades previas a cada clase
- Realizar actividades durante cada clase
- Realizar actividades después de clase
- Participar en incentivos de cátedra con evaluación de pares
- Participar en simulacros de evaluación de cátedra con evaluación de pares
- Estudio personal
- Resolución de dudas entre pares/ayudante/profesora



## Laboratorio

- Realizar actividades previas a cada clase
- Realizar actividades durante cada clase (con entrega obligatoria)
- Realizar actividades después de clase
- Tareas computacionales individuales (implementación e informe)



# Recomendaciones de estudiantes de 01/2023

No subestimar el tiempo de las tareas de Laboratorio

trate de tomarlo con ramos no tan pesados (ejemplo paradigmas).

Realizar las ejercicios propuestos en clases de cátedra, en pseudocódigo y también la implementación de estos en C.

dedique tiempo a esta asignatura

Si toman paradigmas y este ramo tengan cuidado con los tiempos porque algunas tareas de este ramo requieren tiempo y las de paradigmas también requieren un montón de tiempo, asique cuidado con eso.

en cátedra preste atención a lo importante de los TDAs (funciones y posibles implementaciones) y que haga los ejercicios propuestos, para laboratorio que lleve pseudocódigos y los aplique (ayuda tanto para mejorar pseudocódigos como para realizar el laboratorio)

Que vaya a las clases e intente siempre implementar los algoritmos vistos de distintas formas y generar un análisis de estos, así no te agarran tan desprevenido las tareas de la asignatura, ni los pequeños laboratorios.

Considero que un aspecto fundamental a recordar en cátedra es la asistencia a las clases, así como también la resolución de dudas con la profesora, intentando ir un nivel más allá para poder evidenciar que los conceptos enseñados efectivamente han sido captados y aprendidos

mejorar en cuanto a la rapidez y eficiencia para poder realizar soluciones en pseudocódigo para los problemas en muy poco tiempo (Para las pruebas)

presente atención en clases

tener una base sobre cómo funciona el lenguaje de programación C

Practicar, practicar y practicar