

# Estructura de datos y algoritmos

Rodrigo Alvarez

[rodrigo.alvarez2@mail.udp.cl](mailto:rodrigo.alvarez2@mail.udp.cl)

**¿Por qué estudiar sobre estructura de datos y algoritmos ?**

## ¿Por qué estudiar sobre estructura de datos y algoritmos ?

- Resolución de problemas: Muchos problemas de programación y algoritmos se pueden resolver más fácilmente utilizando estructuras de datos adecuadas.

## ¿Por qué estudiar sobre estructura de datos y algoritmos ?

- Resolución de problemas: Muchos problemas de programación y algoritmos se pueden resolver más fácilmente utilizando estructuras de datos adecuadas.
- Organización: Las estructuras de datos te permiten organizar y almacenar datos de manera adecuada para acceder a ellos y manipularlos de manera eficiente.

# ¿Por qué estudiar sobre estructura de datos y algoritmos ?

- Resolución de problemas: Muchos problemas de programación y algoritmos se pueden resolver más fácilmente utilizando estructuras de datos adecuadas.
- Organización: Las estructuras de datos te permiten organizar y almacenar datos de manera adecuada para acceder a ellos y manipularlos de manera eficiente.
- Abstracción: Proporcionan una abstracción para trabajar con datos. Permiten pensar en términos de conceptos abstractos (listas, pilas, colas, árboles, grafos, etc.) en lugar de preocuparte por los detalles de implementación subyacentes.

# ¿Por qué estudiar sobre estructura de datos y algoritmos ?

- Resolución de problemas: Muchos problemas de programación y algoritmos se pueden resolver más fácilmente utilizando estructuras de datos adecuadas.
- Organización: Las estructuras de datos te permiten organizar y almacenar datos de manera adecuada para acceder a ellos y manipularlos de manera eficiente.
- Abstracción: Proporcionan una abstracción para trabajar con datos. Permiten pensar en términos de conceptos abstractos (listas, pilas, colas, árboles, grafos, etc.) en lugar de preocuparte por los detalles de implementación subyacentes.
- Eficiencia: Al comprender cómo funcionan diferentes estructuras de datos y cuándo utilizarlas, el elegir la adecuada puede mejorar significativamente el rendimiento de los algoritmos y programas.



**acm** International Collegiate  
Programming Contest



### Software Engineering

Microsoft · Gran Santiago, Región Metropol...

[Solicitar](#)[Guardar](#)

...

#### Qualifications

##### Required

- 4 years experience delivering quality software within a professional organization.
- 3 years programming in one or more of C#, C, C++, Java
- Experience with RESTful APIs, services and cloud storage technologies
- Understanding of **data structures, algorithms**, threading, synchronization.
- You are an ambitious teammate who is passionate about learning, sharing knowledge, and making your teammates successful.
- You are creative and have strong design, development and collaboration skills.

### Graduate 2024 Software Engineer I B...

Uber · Gran Santiago, Región Metropolitana ...

[Solicitar](#)[Guardar](#)

...

#### Basic qualifications :

- Knowledge of **data structures, algorithms**, programming and associated programming languages and frameworks. (e.g. C, C++, Java, Python or Golang)
- Advanced English.
- Recent bachelor graduate in Computer Science, Engineering, or a related technical field concluded from July/2023 to June/2024.

### Software Engineer III, Full Stack, You...

Google · San Bruno, CA (Presencial)

[Solicitar](#)[Guardar](#)

...

#### Minimum qualifications:

- Bachelor's degree or equivalent practical experience.
- 2 years of experience with software development in one or more programming languages, or 1 year of experience with an advanced degree in an industry setting.
- 2 years of experience with **data structures or algorithms** in either an academic or industry setting.
- 2 years of experience with full stack development, across back-end such as Java, Python, GO, or C++ codebases, and front-end experience including JavaScript or TypeScript, HTML, CSS or equivalent.

### Backend Engineer – Backstage Data ...

Spotify · Nueva York, NY

[Solicitar](#)[Guardar](#)

...

#### Who You Are

- You excel in breaking down complex technical problems and iterating quickly in an environment where engineers are empowered to take ownership of solutions from ideation to production.
- Your experience includes shipping high-impact code and contributing to data platform or developer experience projects.
- You have strong skills in Java and TypeScript, and a solid understanding of systems design, **data structures, and algorithms**.



# Unidades Temáticas

# Bloque 1

## Introducción a Java y estructuras básicas

- Introducción a java y librería estándar
- Algoritmos iterativos y recursivos
- Análisis de complejidad algorítmica
  - Empírico
  - Teórico
- Árboles de recursión y memoización
- Listas
- Pilas y colas

## Bloque 2:

### Ordenamiento y estructuras de datos avanzadas

- Ordenamiento en la stdlib y aplicaciones
- Búsqueda binaria
- Ordenamiento en tiempo cuadrático
- Merge sort
- Priority queues
- Set, Map y Hash tables
- Árboles binarios de búsqueda

## Bloque 3:

### Grafos

- Definición de grafos
- Representación de grafos
- Recorridos en grafos

# Evaluaciones

```
cantidad_solemnes = 2  
cantidad_controles ~= 5  
cantidad_labs ~= 5
```

## fechas

solemne 1: semana 25/09 - 01/10  
solemne 2: semana 27/11 - 03/12  
examen: semana 11/12 - 20/12

## fechas tentativas:

controles: semanas del 26/08, 09/09, 21/10, 04/11, 18/11  
labs: semanas del 19/08, 02/09, 14/10, 28/10, 11/11

# Ponderaciones

- NP: nota de presentación
- NC: nota de controles
- NL: nota de laboratorios
- NS: nota de solemnes

$$NP = 0.3 * NC + 0.3 * NL + 0.4 * NS$$

$$\text{eximido} = NP \geq 5.0 \ \&\& \ NC \geq 4.0 \ \&\& \ NL \geq 4.0 \ \&\& \ NS \geq 4.0$$

- NE: nota de examen
- NF: nota final

$$NF = 0.7 * NP + 0.3 * NE$$

# Recursos

## Cursos

- [Java for Beginners](#)
- [MIT 6.006 Introduction to Algorithms, Spring 2020](#)
- [Kevin Wayne site](#)

## Plataformas

- [Codewars](#)
- [LeetCode](#)
- [HackerRank](#)
- [Codeforces](#)

## Bibliografía

- R. Sedgewick, K. Wayne, Algorithms. Addison-Wesley, 2014.
- Robert Sedgewick, Kevin Wayne - Introduction to Programming in Java (2017)
- J. Kleinberg and E. Tardos, Algorithm design. Boston, Mass: Pearson/Addison-Wesley, 2006.
- S.S. Skiena, M.A. Revilla. Programming Challenges. Springer Verlag, 2003.
- Cormen, Thomas H.; Leiserson, Charles E.; Rivest, Ronald L.; Stein, Clifford (2009) [1990]. Introduction to Algorithms (3rd ed.). MIT Press and McGraw-Hill.



**THANK YOU FOR  
YOUR ATTENTION**



**Q&A SECTION NOW**