

Estructura de datos y algoritmos

Rodrigo Alvarez

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



Unidades Temáticas

Bloque 1

Introducción a Java y estructuras básicas

- Introducción a java y librería estándar
- Notación asintótica
- Listas/Pilas/Colas
- Recursión, call stack
- Algoritmos de ordenamiento
- Algoritmos de búsqueda

Bloque 2:

Estructuras de datos avanzadas

- Árboles
- Set y Map
- Grafos
- Algoritmos de búsqueda y recorrido

Bloque 3:

Bonus

- Programación dinámica
- Búsqueda informada

Evaluaciones

```
cantidad_solemnes = 2  
cantidad_controles ~= 4  
cantidad_labs ~= 5
```

fechas

solemne 1: semana 24/04 - 30/04

solemne 2: semana 24/06 - 28/06

examen: semana 08/07 - 17/07

fechas tentativas:

controles: 25/03, 08/04, 27/05, 10/06

labs: 18/03, 01/04, 15/04, 13/05, 03/06

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

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

Cambio de día catedra

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