

Curso de nivelación de algoritmos

Clase 1



Curso de nivelación de algoritmos

Docentes: Herman Schinca y Hernán Czemerinski

Clase de hoy:

Cuestiones administrativas: horarios, mails, web, etc.

Objetivos, evaluación, bibliografía.

Repaso superficial del contenido de toda la materia.

Temas introductorios de programación.

Clase práctica: setup de cuentas, Linux, Python.

Curso de nivelación de algoritmos

Docentes: Herman Schinca (hschinca@dc.uba.ar)
Hernán Czemerinski (hczemer@dc.uba.ar)

Horarios y lugar:

Jueves 13 y viernes 14 de 18 a 22 horas.

Lunes 17 al miércoles 19 de 18 a 22 horas.

Laboratorios 4 y 5.

Sitio web: <http://datamining.dc.uba.ar/campus>

Curso de nivelación de algoritmos

Labo 4:

Usuarios: [introalgo-001](#) a [introalgo-050](#)

Contraseña: [intro2018](#)

Labo 5:

Usuarios: [introalgo-051](#) a [introalgo-100](#)

Contraseña: [intro2018](#)

Curso de nivelación de algoritmos

Objetivos:

Al finalizar el curso, se espera que el alumno pueda resolver, utilizando algoritmos, problemas de tamaño pequeño a mediano.

En particular, se apunta a que los alumnos dominen elementos básicos de: programación, técnicas algorítmicas y de verificación, noción de complejidad computacional y tipos abstractos de datos.

Curso de nivelación de algoritmos

Modo de evaluación:

Un examen final.

Bibliografía:

- Elkner, Downey & Meyers, "How to Think Like a Computer Scientist", <http://www.openbookproject.net/thinkcs/python/english2e/>
- Tutorial online de Python, <http://docs.python.org/tutorial/>
- Cormen, "Introduction to Algorithms", MIT Press, 2009.
- Aho, Hopcroft y Ullman, "Data Structures and Algorithms", Addison-Wesley, 1983.

Curso de nivelación de algoritmos

¿Qué es programar?

- Programar \neq Conocer un lenguaje de programación.
- Especificación formal, corrección, eficiencia, modularidad, usabilidad, adaptabilidad, ...

¿Qué lenguajes de programación conocen?

Curso de nivelación de algoritmos

1) Elementos básicos de programación

- Tipos de datos: enteros, reales, strings, etc.
- Variables y expresiones.
- Instrucción: asignación, condicional, ciclo.
- Estado de un programa.
- Funciones, pasaje de parámetros.

2) Estructuras de datos

3) Algoritmos de búsqueda y ordenamiento

4) Noción de complejidad

5) Técnicas algorítmicas

Curso de nivelación de algoritmos

1) Elementos básicos de programación

2) Estructuras de datos

- Listas
- Pilas
- Colas
- Conjuntos
- Diccionarios

3) Algoritmos de búsqueda y ordenamiento

4) Noción de complejidad

5) Técnicas algorítmicas

Curso de nivelación de algoritmos

1) Elementos básicos de programación

2) Estructuras de datos

3) Algoritmos de búsqueda y ordenamiento

- Buscar un elemento en una lista.
- Ordenar los elementos de una lista.

4) Noción de complejidad

5) Técnicas algorítmicas

Curso de nivelación de algoritmos

1) Elementos básicos de programación

2) Estructuras de datos

3) Algoritmos de búsqueda y ordenamiento

4) Noción de complejidad

- ¿Cuánto tiempo computacional demanda hallar la solución de un problema?

5) Técnicas algorítmicas

Curso de nivelación de algoritmos

- 1) Elementos básicos de programación**
- 2) Estructuras de datos**
- 3) Algoritmos de búsqueda y ordenamiento**
- 4) Noción de complejidad**
- 5) Técnicas algorítmicas**
 - Divide & Conquer
 - Backtracking

Curso de nivelación de algoritmos

Pregunta frecuente:

- ¿Qué lenguaje de programación vamos a usar?

Curso de nivelación de algoritmos

Pregunta frecuente:

- ¿Qué lenguaje de programación vamos a usar?

Respuesta corta:

- Python.

Curso de nivelación de algoritmos

Pregunta frecuente:

- ¿Qué lenguaje de programación vamos a usar?

Respuesta corta:

- Python.

Respuesta larga:

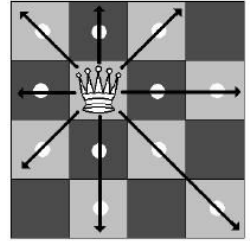
- No importa demasiado. Lo que más importa son los conocimientos básicos de programación, que son comunes a la mayoría de los lenguajes.

Curso de nivelación de algoritmos

Dos problemas para pensar:

Colocar 8 reinas en un tablero de ajedrez sin que se amenacen.
(Tablero: 8x8.)

<http://www.brainmetrix.com/8-queens/>

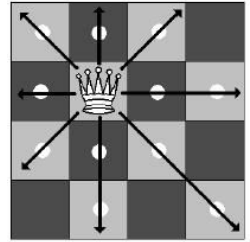


Curso de nivelación de algoritmos

Dos problemas para pensar:

Colocar 8 reinas en un tablero de ajedrez sin que se amenacen.
(Tablero: 8x8.)

<http://www.brainmetrix.com/8-queens/>



Torre de Hanoi:

Mover N discos de la estaca 1 a la 3.

Mover de a un disco por vez.

No se puede colocar un disco sobre otro de menor tamaño.

<http://www.webgamesonline.com/towers-of-hanoi/>

