



- ### Objetivos
- general
 - resolver problemas
 - específico
 - escribir programas que resuelvan problemas (en lenguajes de programación Python y Matlab)
 - de largo plazo (propósitos)
 - razonamientos algorítmico y lógico
 - capacidades de abstracción y modelamiento
 - habilidad general para resolver problemas

- ### Contenidos
1. Fundamentos de programación
 2. Programación orientada a objetos
 3. Listas y tablas de valores
 4. Estructuras de datos
 5. Computación científica

- ### Evaluación
- NF (nota final): $70\%NC + 30\%NT$ ($NC, NT \geq 4$)
 - NT (nota tareas): promedio 5 tareas (1 por cap)
Si $3 \leq NT < 4$ y $NC \geq 4$, $NF = I +$ tarea(s) extra(s)
 - NC (nota control): prom ponderado controles
- | control | contenidos | fecha | pond |
|---------|------------|-----------|------|
| 1 | 1 | Semana 4 | 20 % |
| 2 | 2 | Semana 8 | 20 % |
| 3 | 3 y 4 | Semana 13 | 20 % |
| examen | 1 a 5 | ? | 40 % |

- ### Ejercicios
- se evalúan entre 1 y 4 ($\% = 0, \leq 33, \leq 66, \leq 100$)
 - se considera promedio de los 20 mejores (de 27)
 - si tiene menos de 20, $NF = I +$ tarea(s) extra(s)
 - sirven para eximirse de examen de acuerdo a la siguiente tabla:
- | promedio c1,c2,c3 | promedio ejercicios |
|-------------------|---------------------|
| 5,5 | cualquiera |
| 5,4 | ≥ 1.5 |
| 5,3 | ≥ 2.0 |
| 5,2 | ≥ 2.5 |
| 5,1 | ≥ 3.0 |
| 5,0 | ≥ 3.5 |

- ### Bibliografía
- Apuntes del curso
 - clase a clase (quedan disponibles en U-cursos)
 - “Libros” complementarios (disponibles en Internet)
 - “Think Python”, Downey
 - “Python for Informatics”, Severance
 - “Introduction to Programming using Python”, Schuerer
 - “Python para todos”, Gonzalez Duque
 - “Python Tutorial” (+avanzado); Guido van Rossum (creador de Python)
 - Tutor en línea de python <http://people.csail.mit.edu/pgbovine/python/tutor.html#mode=edit>