



UNIVERSIDAD NACIONAL DE INGENIERÍA
FACULTAD DE CIENCIAS
ESCUELA PROFESIONAL DE CIENCIA DE LA COMPUTACIÓN

SÍLABO

INFORMACIÓN GENERAL

ASIGNATURA	:	INTRODUCCIÓN A LA PROGRAMACIÓN
CÓDIGO	:	CC102
CRÉDITOS	:	02 (DOS)
PRE-REQUISITO	:	CC101 – INTRODUCCIÓN A LA CIENCIA DE LA COMPUTACIÓN
CONDICIÓN	:	OBLIGATORIO
HORAS POR SEMANA	:	04 (TEORÍA: 01, PRÁCTICA: 02, SEMINARIO: 01)
SISTEMA DE EVALUACIÓN	:	G

OBJETIVO

Este curso introducirá a los participantes en los conceptos fundamentales de este arte. Los tópicos incluyen tipos de datos, estructuras de control, funciones, listas, recursividad y la mecánica de la ejecución, prueba y depuración.

El curso también ofrecerá una introducción al contexto histórico y social de la informática y una revisión del ámbito de esta disciplina.

PROGRAMA ANALÍTICO

1. Lenguajes de programación

I: lenguajes naturales y artificiales. II: Elementos y evolución. III: Diseño e implementación. IV: Taxonomía, paradigmas y popularidad.

2. Sistemas de tipos

I: Características. II: Verificación estática, dinámica, fuerte y débil. III: Lenguajes type-safe y type-unsafely.

3. Funciones y Recursividad

I: Definiciones. II: Características. III: Ámbito de variables. IV: Parâmetros por valor y por referencia.

4. Estructuras de datos fundamentales 1

I: Arreglos. II: Registros.

5. **Punteros 1**

I: Declaraciones. II: Operadores. III: Paso de parámetros.

6. **Punteros 2**

I: Expresiones y aritmética de punteros.

7. **Memory Allocation**

I: Código fuente, ejecutable y proceso. II: memoria virtual. III: Allocation y Deallocation de memoria. IV: problemas de memoria con punteros.

8. **Estructuras de datos fundamentales 2**

I: Listas enlazadas. II: Listas doblemente enlazadas.

9. **Fundamentos de tipos abstractos de datos 1**

I: Pilas. II: Colas.

10. **Fundamentos de tipos abstractos de datos 2**

I: Árboles. II: Árboles binarios. III: Árboles binarios de búsqueda.

11. **Archivos**

I: Métodos de acceso. II: Secuenciales, binarios, haced. III: Directorios. IV: Binarios y ASCII.

12. **Portabilidad**

I: Definición. II: Heurísticas generales. III: Hardware heterogéneo. IV: SO heterogéneos. V: Compiladores y librerías heterogéneas.

13. **Optimización**

I: Desempeño de los programas. II: Optimizando desempeño de los compiladores. III: Efectos colaterales. IV: Adición y multiplicación. V: Llamada a funciones

14. **Usabilidad (módulos y bibliotecas)**

I: Linkeo dinámico. II: Linkeo estático.

BIBLIOGRAFÍA

1. Deitel, P. C++ How to program (6th edición). Prentice Hall. 2007.
2. Forouzan, B., Mosharraf, F., Foundations of Computer Science. Cengage Learning Business Press, 2008.
3. Bird, R. J. Introduction to Functional Programming using Haskell. Prentice-Hall Series in Computer Science. Prentice-Hall Europe, London, UK, 2nd edition. 1998.
4. Paz-Valderrama, A. Apuntes: Programación funcional en Haskell. Sociedad Peruana de Computación. 2005.
5. Thompson, S. The Craft of Functional Programming, 2E. Addison Wesley, 1999.