| | |
|---|---|
| **Nombre de la materia:** | Paradigmas de Programación |
| **Clave:** | IA7300-T |
| **No. de horas/semana:** | 3 |
| **Total de horas:** | 48 |
| **No. de créditos:** | 6 |
| **Prerrequisitos:** | Ingeniería de Programación (CI7101-T) |

**Objetivo general:** Este curso introduce al estudiante a los diferentes tipos de lenguajes de programación contemporáneos. A lo largo de la historia de la computación, se ha desarrollado un gran número de lenguajes de programación, cada uno con diferentes objetivos en mente. Sin embargo, estos lenguajes se pueden clasificar, a groso modo, en cuatro familias: Imperativos, Orientados a Objetos, Lógicos y Funcionales. Dentro de este estudio, se incluyen los conceptos fundamentales que le proporcionan a un programador las herramientas necesarias para poder hacer un uso eficiente y con conocimiento de cualquier lenguaje de programación. Estos conceptos incluyen chequeo de tipos, administración de memoria, scoping, paso de parámetros, polimorfismo, etc., los cuales serán incluidos en los diversos temas, conforme se presenten en los diferentes lenguajes analizados. Esta estructura le proporciona al profesor la posibilidad de elegir los lenguajes con los cuales ilustrará los diferentes conceptos.

**Contribución a los atributos de egreso y su nivel de aportación**

- **AE1.** Aplicar los conocimientos de ingeniería adquiridos durante sus estudios para elaborar proyectos de ingeniería que resuelvan problemas específicos. **(Medio)**
- **AE2.** Identificar, formular y resolver problemas de ingeniería mediante un pensamiento crítico y asertivo, basados en los principios de ciencias básicas e ingeniería. **(Avanzado)**
- **AE3.** Presentar y defender su trabajo en diversos foros, tanto académicos como profesionales. **(Inicial)**
- **AE4.** Intercambiar su conocimiento y puntos de vista con profesionales del área e integrarse en equipos de trabajo multidisciplinarios. **(Avanzado)**
- **AE5.** Respetar su entorno social y disciplinar, enmarcado siempre por valores humanos y de ética profesional, con una actitud creativa y positiva para enfrentar nuevos retos. **(Inicial)**

**Programa sintético**

1. Historia y evolución de los lenguajes de programación ..... 3 hrs.
2. Lenguajes Imperativos ..... 9 hrs.
3. Lenguajes Orientados a Objetos ..... 9 hrs.
4. Lenguajes Funcionales ..... 9 hrs.
5. Lenguajes Lógicos ..... 9 hrs.
6. Lenguajes Script ..... 9 hrs.

Total: 48 hrs.

**Programa desarrollado**

1. Historia y evolución de los lenguajes de programación ..... 3 hrs.
   1.1 Línea de tiempo y el desarrollo de los lenguajes de programación

- 1.2 Familias de lenguajes de programación
- 1.3 Niveles de abstracción
- 1.4 Paradigmas de programación
- 1.5 Lenguajes de programación e ingeniería de software

2. Lenguajes Imperativos ..... 9 hrs.
   - 2.1 Variables y asignación
   - 2.2 Goto, comandos no estructurados
   - 2.3 Programación estructurada
   - 2.4 Composición secuencial, condicionales y ciclos
   - 2.5 Escapes y excepciones
   - 2.6 Variables globales y efectos colaterales
   - 2.7 Aliases y apuntadores perdidos (dangling pointers)
   - 2.8 Subrutinas, subprogramas, procedimientos y funciones
   - 2.9 Paso de parámetros (valor, referencia, valor-resultado y nombre)
   - 2.10 Ligas estáticas y dinámicas a registros de activación
3. Lenguajes Orientados a Objetos ..... 9 hrs.
   - 3.1 Encapsulación (contra quien nos protegemos)
   - 3.2 Miembros datos y métodos
   - 3.3 Control de acceso (miembros públicos y privados)
   - 3.4 Herencia: subclases y superclases
   - 3.5 Relaciones contiene y es-un
   - 3.6 Control de acceso (miembros protegidos)
   - 3.7 Polimorfismo (cambio de tipos)
4. Lenguajes Funcionales ..... 9 hrs.
   - 4.1 Cálculo lambda
   - 4.2 Expresiones simbólicas
   - 4.3 Datos = programas, el enfoque de Von Newmann
   - 4.4 Let y let*, condicionales e iteración
   - 4.5 Mapeos
   - 4.6 Recursividad de Cola, argumentos colectores
   - 4.7 Macros, paso de parámetros por nombre
5. Lenguajes Lógicos ..... 9 hrs.
   - 5.1 Lógica de Predicados (repaso breve)
   - 5.2 Relaciones
   - 5.3 Queries
   - 5.4 Reglas o programas
   - 5.5 Asignación vs. Instanciamiento
   - 5.6 Backtracking
   - 5.7 Cuts y control de búsqueda

6. Lenguajes Script ..... 9 hrs.

6.1 Justificación para los lenguajes script
6.2 Relación de los lenguajes script con otros tipos de lenguajes
6.3 Portabilidad de lenguajes script
6.4 Compilación e interpretación de lenguajes script

**Bibliografía básica:**

- Friedman, D. P.; Wand, M.; Heynes, C. T., Essentials Of Programming Languages, The MIT Press, 1992
- Sethi, R., Programming Languages, Concepts and Constructs, Addison-Wesley Publishing Company, 1989

**Bibliografía complementaria:**

- Budd, T., An Introduction To Object-Oriented Programming, Addison-Wesley Publishing Company, 1991.
- Field, A. J.; Harrison, P. G., Functional Programming, Addison-Wesley Publishing Company, 1989.
- Friedman, L. W., Comparative Programming Languages, Generalizing The Programming Function, Prentice Hall,Inc., 1991.
- Kogge, P. M., The Architecture of Symbolic Computers, McGraw-Hill Incorporated, 1991.
- Tucker, A. B., Jr., Lenguajes De Programación, Segunda Edición, McGraw-Hill,, España, 1987.

**Metodologías de enseñanza-aprendizaje:**

- Revisión de conceptos, análisis y solución de problemas en clase ( X )
- Lectura de material fuera de clase ( X )
- Ejercicios fuera de clase (tareas) ( X )
- Investigación documental ( X )
- Elaboración de reportes técnicos o proyectos ( X )
- Prácticas de laboratorio en una materia asociada ( X )

**Metodologías de evaluación:**

- Asistencia ( X )
- Tareas ( X )
- Elaboracion de reportes técnicos o proyectos ( X )
- Exámenes de academia o departamentales ( X )

**Revisores:**

Programa propuesto por Dr. Juan José Flores Romero y actualizado por M.C. Luis Eduardo Gamboa Guzmán.