

Anexo II

INSTITUTO: Tecnología e Ingeniería

CARRERA/S: Licenciatura en Informática

MATERIA: Características de Lenguajes de Programación

NOMBRE DEL RESPONSABLE DE LA ASIGNATURA: Pablo Miguel Angel Pandolfo

EQUIPO DOCENTE: -----

CUATRIMESTRE: 1ro.

AÑO: 5to.

PROGRAMA N°: 39 (Aprob. Por Cons.Directivo fecha XX)

Instituto/s: Tecnología e Informática

Carrera/s: Licenciatura en Informática.

Nombre de la materia: Características de Lenguajes de Programación

Responsable de la asignatura y equipo docente: Pablo Miguel Angel Pandolfo

Cuatrimestre y año: 1er. cuatrimestre, 5to. año.

Carga horaria semanal: 4 horas

Programa N°: 39

Código de la materia en SIU: 780

Características de Lenguajes de Programación

1. Fundamentación

El conocimiento profundo de los conceptos intrínsecos de los lenguajes de programación permite adquirir herramientas para construir criterios de evaluación de los lenguajes. Esto hace posible consolidar los conocimientos sobre los lenguajes ya conocidos e incorporar elementos para abordar los nuevos.

Como formación complementaria se alienta y evalúa el trabajo colaborativo y la capacidad de presentar y exponer trabajos en forma escrita y oral.

2. Propósitos y/u objetivos

Objetivos:

Que el estudiante:

- Analice, compare y evalúe los conceptos subyacentes de los Lenguajes de Programación en los distintos Paradigmas.
- Adquiera la capacidad de evaluar lenguajes de programación desde distintos puntos de vista, ya sea como su diseñador, implementador o como usuario del lenguaje.

3. Programa sintético:

- Lenguajes según su modelo de cómputo: imperativo, funcional, objetos, lógico. Estructuras de datos en los diferentes paradigmas. Lenguajes según sus características: Lenguajes tipados y no tipados. Mecanismos de binding (estático y dinámico). Mecanismos de pasaje

de parámetros (valor, referencia, nombre, otros). Formas de llevar a cabo la ejecución (compilación, interpretación, máquinas virtuales). Formas de administración de memoria (explícita y garbage collection). Lenguajes según su propósito: generales, de dominio específico, de scripting. Lenguajes con semánticas operacional, denotacional, axiomática. Lenguajes de Programación: Entidades y ligaduras. Sistema de Tipos, Niveles de Polimorfismo. Encapsulamiento y Abstracción. Conceptos de Intérpretes y Compiladores. Criterios de Diseño y de Implementación de Lenguajes de Programación. Nociones básicas de semántica formal.

4. Programa analítico

4.1 Organización del contenido:

Unidad 1: Introducción.

Razones para estudiar los Lenguajes de Programación. Definición de Lenguaje de Programación. Historia. Abstracción. Clasificación. Descripción. Diseño.

Unidad 2: Paradigmas.

Imperativo. Funcional. Orientado a Objetos. Lógico. Multiparadigma. Características. Comparación de lenguajes en diferentes paradigmas. Efectos de estos conceptos tienen lugar en los paradigmas de solución de problemas.

Unidad 3: Sintaxis y Semántica.

Sintaxis de los Lenguajes de Programación. Estructura léxica de los Lenguajes de Programación. Gramáticas libres de contexto. Notación BNF. Árboles sintácticos. Ambigüedades, asociatividad y precedencia. Diagramas sintácticos. Semántica de los Lenguajes de Programación. Atributos, vínculos y funciones semánticas. Declaraciones, bloques y alcance. La tabla de símbolos. Asignación, tiempo de vida y ambiente. Variables y constantes. Semántica formal.

Unidad 4: Tipos de Datos.

Tipos de datos atómicos y estructurados. Tipos abstractos. Implementación de datos. Sistemas de tipos. Lenguajes seguros y fuertemente tipados. Seguridad en el manejo de tipos. Encapsulamiento y abstracción. Equivalencia de Tipos de Datos. Conversión de Tipos de Datos.

Unidad 5: Control de Ejecución.

Evaluación de expresiones. Sentencias condicionales. Bucles. Excepciones. Subprogramas. Paso de parámetros. Ambientes de Ejecución.

4.2 Bibliografía y recursos obligatorios:

Louden, K. C. (2011). Programming languages: principles and practices.

Pratt, T (2001). Programming Languages. Design and Implementation, 4ta. edición, Prentice Hall.

Sethi, R. (1996). Programming languages: concepts and constructs, 2da. edición, Addison – Wesley.

4.3 Bibliografía optativa:

Libros.

Tucker, A. (2006). Programming languages, McGraw Hill

5. Metodologías de enseñanza:

Las clases serán en modalidades teórica, práctica y teórico-práctico dependiendo del tema a desarrollar.

En las clases teóricas se reforzará con un material de lectura (artículos, capítulos de libros y links a sitios de internet) y en algunos casos con cuestionarios para realizar a través del campus.

Se dará a los/as alumnos/as guías de trabajos prácticos con casos prácticos ejercicios para realizar con el objetivo de asentar los conceptos trabajados en clase. Algunos de estos ejercicios formarán parte de su evaluación.

Plan de trabajo en el campus:

El Campus Virtual es un espacio fundamental para el desarrollo de la asignatura. En el aula virtual se propondrá material educativo, apuntes de clase, bibliografía, así como también el programa y cronograma de la asignatura y las guías de trabajos prácticos.

6. Actividades de investigación y extensión (si hubiera)

En la materia no se desarrollan actividades de investigación en el marco de un proyecto de investigación. No obstante, algunos puntos de las guías de trabajos prácticos requieren que los/as alumnos/as investiguen para poder resolverlos.

7. Evaluación y régimen de aprobación

7.1 Aprobación de la cursada

Para aprobar la cursada y obtener la condición de regular, el régimen académico establece que debe obtenerse una nota no inferior a cuatro (4) puntos. Todas las instancias evaluativas deberán tener una instancia de recuperatorio. Podrán acceder a la administración de esta modalidad solo aquellos y aquellas estudiantes que hayan obtenido una nota inferior o igual a 6 (seis) puntos en el examen parcial.

Siempre que se realice una evaluación de carácter recuperatorio, la calificación que los/as estudiantes obtengan reemplazará la calificación obtenida en el examen que se ha recuperado y será la considerada definitiva a los efectos de la aprobación.

El alumno deberá poseer una asistencia no inferior al 75% en las clases presenciales.

En cuanto a las cursadas de materias virtuales se requerirá que el estudiante ingrese al aula virtual como mínimo una vez por semana.

7.2 Aprobación de la materia

La materia puede aprobarse por promoción, evaluación integradora, examen final o libre.

Promoción directa: tal como lo establece el art°17 del [Régimen Académico](#), para acceder a esta modalidad, el/la estudiante deberá aprobar la cursada de la materia con una nota no inferior a siete (7) puntos, no obteniendo en ninguna de las instancias de evaluación parcial menos de seis (6) puntos, sean evaluaciones parciales o recuperatorios. El promedio estricto resultante deberá ser una nota igual o superior a siete (7) sin mediar ningún redondeo.

Evaluación integradora: tal como lo establece el art°18 del [Régimen Académico](#), podrán acceder a esta evaluación aquellos estudiantes que hayan aprobado la cursada con una nota de entre cuatro (4) y seis (6) puntos.

La evaluación integradora tendrá lugar por única vez en el primer llamado a exámenes finales posterior al término de la cursada. Deberá tener lugar en el mismo día y horario de la cursada y será administrado, preferentemente, por el/la docente a cargo de la comisión. Se aprobará tal instancia con una nota igual o superior a cuatro (4) puntos, significando la aprobación de la materia.

La nota obtenida se promediará con la nota de la cursada.

Examen final: Instancia destinada a quienes opten por no rendir la evaluación integradora o hayan regularizado la materia en cuatrimestres anteriores. Se evalúa la totalidad de los contenidos del programa de la materia y se aprueba con una calificación igual o superior a cuatro (4) puntos. Esta nota no se promedia con la cursada.

7.3 Criterios de calificación

El sistema normal de evaluación consistirá en 2 (dos) exámenes parciales con recuperatorios, según el cronograma previsto, de la totalidad de la materia descrita en el programa. Los mismos se realizarán en las fechas que, a tal efecto, se establezcan en el cronograma. El primer parcial es teórico-práctico e individual y el segundo parcial es un trabajo integrador desarrollado por los/as alumnos/as de manera grupal y se debe defender por todos los integrantes del grupo. Además, se considera como parte de la evaluación de la cursada el desarrollo de la guía de trabajos prácticos que los/as alumnos/as deben presentar de manera grupal al final de la cursada de la asignatura.

8. Cronograma

CLASE	UNIDAD	TEMAS	ACTIVIDADES	MODALIDAD
1	Introducción	Presentación de la materia. Conceptos fundamentales.	Preguntas teóricas	Virtual. Sincrónica
2	Introducción	Historia y Clasificación	Ejercicios de clasificación de Lenguajes de Programación	Virtual. Sincrónica
3	Paradigmas	Imperativo. Funcional. Objetos. Lógico	Presentaciones grupales	Virtual Sincrónica
4	Paradigmas	Comparación de lenguajes	Ejercicios de comparación. Resolución de problemas en diferentes paradigmas	Virtual Sincrónica
5	Sintaxis Semántica	y Sintaxis	Ejercicios de Sintaxis	Virtual Sincrónica
6	Sintaxis Semántica	y Sintaxis	Ejercicios de Sintaxis	Virtual Sincrónica
7	Sintaxis Semántica	y Semántica	Ejercicios de Semántica	Virtual Sincrónica
8	Examen – Primer Parcial			Presencial
9	Tipos de Datos	Tipos de datos	Devolución de parciales Ejercicios de Tipos de datos	Virtual Sincrónica
10	Tipos de Datos	Equivalencia	Ejercicios de equivalencias de tipos de datos	Virtual Sincrónica
11	Tipos de Datos	Conversión	Ejercicios de conversión de tipos de datos	Virtual Sincrónica
12	Control de Ejecución	Expresiones, condicionales y bucles	Ejercicios varios	Virtual Sincrónica
13	Control de Ejecución	Excepciones	Ejercicios de excepciones	Virtual Sincrónica
14	Control de Ejecución	Subprogramas	Ejercicios de subprogramas	Virtual Sincrónica

15	Examen Segundo Parcial			Presencial
16	Recuperatorios		Finalización del cuatrimestre. Indicaciones para la evaluación integradora	Presencial