

Programa de PROGRAMACIÓN IMPERATIVA

1. NOMBRE DE LA UNIDAD CURRICULAR

Programación imperativa

2. CRÉDITOS

10 créditos

3. OBJETIVOS DE LA UNIDAD CURRICULAR

El objetivo de esta unidad curricular es presentar al estudiante los conceptos básicos de programación, dentro del paradigma de la programación imperativa. Se introducirán los elementos básicos de un lenguaje de programación y se trabajará en el diseño y la implementación de algoritmos simples. A nivel de los objetivos de aprendizaje, el estudiante será capaz de:

- Comprender el modelo de computación.
- Diseñar algoritmos para resolver problemas simples.
- Construir pequeños programas o subprogramas utilizando un lenguaje de programación imperativa, llevando adelante tareas de codificación, compilación y depuración de errores.
- Utilizar conceptos básicos de la programación imperativa: identificadores, variables, tipos de datos, estructuras de control, subprogramas.
- Utilizar distintos mecanismos de pasaje de parámetros entre subprogramas.
- Utilizar las estructuras de control más apropiadas para cada problema, especialmente las diferentes estructuras de iteración.
- Diseñar e implementar algoritmos recursivos simples.
- Identificar mejoras a la calidad del código basadas en la aplicación de buenas prácticas de diseño e implementación, y en la ejecución de casos de prueba.
- Aplicar nociones básicas de eficiencia en el diseño de soluciones.

4. METODOLOGÍA DE ENSEÑANZA

La metodología se basa fuertemente en el trabajo del estudiante, articulado en unidades teórico-prácticas. Los estudiantes disponen de bibliografía y materiales en la plataforma EVA.



Durante el curso se dictan 5 horas de clase semanales durante 15 semanas, que incluyen contenidos teóricos y prácticos. Las clases se organizan en forma interactiva, presentando los conceptos en relación con los ejercicios y con ejemplos, y fomentando la participación de los estudiantes. En total se dictan 75 horas de clase. La asistencia a clases podría sustituirse por una cantidad equivalente de horas de estudio de los materiales recomendados en la Bibliografía.

Además de las horas presenciales de clase y/o de estudio de los materiales vía plataforma y libro del curso, el estudiante debe dedicar 75 horas de trabajo personal a la resolución de ejercicios prácticos y de las tareas de laboratorio. Los estudiantes deben realizar trabajos obligatorios en máquina. Para esto se utiliza un lenguaje de programación estructurada fuertemente tipado, con tipado estático.

5. TEMARIO

1. Introducción a la computación
 - a. Modelo de computación y concepto de algoritmo
 - b. Lenguaje de programación. Sintaxis y semántica
 - c. Compilación y ejecución de programas. Tipos de errores
2. Introducción a la programación imperativa
 - a. Estructura de un programa
 - b. Componentes básicos: identificadores, constantes, variables, instrucciones, expresiones, tipos de datos y entrada/salida
 - c. Instrucción de asignación
3. Estructuras de control
 - a. Secuencia
 - b. Selección
 - c. Iteración
 - d. Recursión
4. Tipos de datos
 - a. Tipos elementales
 - b. Tipos estructurados (arreglos, registros)
5. Subprogramas
 - a. Procedimientos y funciones
 - b. Pre y post condiciones
 - c. Pasaje de parámetros



- d. Reglas de alcance
- 6. Algoritmos básicos
 - a. Búsqueda lineal
 - b. Búsqueda binaria
 - c. Ordenación
 - d. Nociones elementales de análisis de algoritmos

6. BIBLIOGRAFÍA

Tema	Básica	Complementaria
Todos los temas	(1,2)	(3)

6.1 Básica

1. Materiales creados por los docentes del curso.
2. Konvalina, John y Stanley Wileman (1987). Programación con Pascal. McGraw-Hill. ISBN 9789684222953.

6.2 Complementaria

3. Grogono, Peter (1984). Programming in Pascal. Addison Wesley. ISBN 0201120704.

7. CONOCIMIENTOS PREVIOS EXIGIDOS Y RECOMENDADOS

7.1 Conocimientos Previos Exigidos: Ninguno.

7.2 Conocimientos Previos Recomendados: Ninguno.

ANEXO A
Para todas las Carreras

A1) INSTITUTO

Instituto de Computación

A2) CRONOGRAMA TENTATIVO

Semana 1	Introducción a la computación
Semana 2	Introducción a la programación imperativa
Semana 3	Secuencia y selección
Semana 4	Iteración
Semana 5	Iteración
Semana 6	Subprogramas
Semana 7	Recursión
Semana 8	Consultas de laboratorio y ejercicios
Semana 9	Tipos de datos
Semana 10	Tipos de datos
Semana 11	Tipos de datos
Semana 12	Algoritmos: búsqueda y ordenación, análisis de algoritmos
Semana 13	Consultas de laboratorio y ejercicios
Semana 14	Consultas de laboratorio y ejercicios
Semana 15	Consultas de laboratorio y ejercicios

A3) MODALIDAD DEL CURSO Y PROCEDIMIENTO DE EVALUACIÓN

La unidad curricular se evalúa mediante pruebas parciales y tareas de laboratorio. Los parciales incluyen ejercicios vinculados al trabajo de laboratorio. A continuación, se presentan las condiciones que deben cumplirse para obtener la suficiencia y la exoneración en el curso; si no se cumplen estos requisitos, el resultado será la reprobación del curso.

La evaluación del curso consiste en:

- Dos pruebas parciales
 - Primer parcial: 40 puntos
 - Segundo parcial: 60 puntos
- Tareas de laboratorio



Resultado de la evaluación:

- Aprobando el laboratorio y obteniendo al menos 25 puntos acumulados entre los dos parciales se aprueba el curso (debe rendirse examen o recurrar).
- Aprobando el laboratorio y obteniendo un puntaje total mayor o igual a 60 entre en los dos parciales, se exonera el examen.
- En caso contrario se reprueba el curso.

El examen incluye un ejercicio vinculado al trabajo de laboratorio del dictado previo.

A4) CALIDAD DE LIBRE

Esta unidad curricular adhiere a la resolución de calidad de libre.

A5) CUPOS DE LA UNIDAD CURRICULAR

Cupos mínimos: no tiene
Cupos máximos: no tiene

APROB. RES. CONSEJO DE FAC. DE ING.
Fecha: 14/07/2026 Expe 060120-000037-26