

PROGRAMACIÓN IMPERATIVA CON LENGUAJE C

Omar Iván Trejos Buriticá



ECOE
EDICIONES

CONTENIDO

INTRODUCCIÓN.....	XV
LECCIÓN 1: LOS PRIMEROS PASOS	19
1.1. ¿Qué es programar?	19
1.2. Paradigma de programación.....	20
1.3. Lenguaje de Programación	20
1.4. Aprender a programar	21
1.5. Lenguaje C.....	22
1.6. Ejercicios propuestos	23
LECCIÓN 2: EL PRIMER PROGRAMA.....	25
2.1. Concepto de función	25
2.2. Primer programa.....	25
2.3. Código completo	28
2.4. Ejercicios propuestos	29
LECCIÓN 3: PROBLEMAS.....	31
3.1. Definición.....	31
3.2. Clasificación.....	32
3.3. Dispositivos de alta tecnología.....	33

3.4. El ser humano	33
3.5. Información	34
3.6. Ejercicios propuestos	34
LECCIÓN 4: METODOLOGÍAS PARA RESOLVER PROBLEMAS	35
4.1. Metodología Polya	35
4.2. Metodología McMaster	37
4.3. Ejercicios propuestos	38
LECCIÓN 5: METODOLOGÍA PARA RESOLVER PROBLEMAS COMPUTABLES	39
5.1. Aproximación	39
5.2. La fase humana	40
5.3. La fase técnica	42
5.4. Ejercicios propuestos	43
LECCIÓN 6: VARIABLES I	45
6.1. Variables	45
6.2. Tipos de datos	46
6.3. Ejercicios propuestos	50
LECCIÓN 7: VARIABLES II	51
7.1. Reglas de las variables	51
7.2. Ejercicios propuestos	55
LECCIÓN 8: OPERADORES	57
8.1. Definición.....	57
8.2. Operadores aritméticos.....	57
8.3. Operadores relacionales	57
8.4. Operadores booleanos	58
8.5. Jerarquía de operadores	59
8.6. Ejercicios propuestos	59
LECCIÓN 9: OPERADORES ARITMÉTICOS	61
9.1. Operadores	62
9.2. Ejercicios propuestos	64
LECCIÓN 10: FUNCIONES DE ENTRADA Y SALIDA PRIMERA PARTE	67
10.1. Ejercicio resuelto	67
10.2. Ejercicios propuestos	70

LECCIÓN 11: FUNCIONES DE ENTRADA Y SALIDA. SEGUNDA PARTE.	71
11.1. Ejercicio	71
11.2. Código	71
11.3. Observaciones	73
11.4. Ejercicios propuestos	73
LECCIÓN 12: UN PROGRAMA COMPLETO I	75
12.1. Enunciado	75
12.2. Objetivo	75
12.3. Algoritmo	75
12.4. Prueba de escritorio	76
12.5. Construcción de funciones	76
12.6. Ejercicios propuestos	79
LECCIÓN 13: UN PROGRAMA COMPLETO II.....	81
13.1. Observaciones	81
13.2. Código	82
13.3. Ejercicios propuestos	84
LECCIÓN 14: CONDICIONALES I.....	85
14.1. Definición.....	85
14.2. Consideraciones	86
14.3. La gran clave	86
14.4. Enunciado	86
14.5. Planteamiento de la solución.....	87
14.6. Ejercicios propuestos	89
LECCIÓN 15: CONDICIONALES II	91
15.1. Construcción del programa.....	91
15.2. Ejercicios propuestos	96
LECCIÓN 16: CONDICIONALES III.....	97
16.1. Solución completa.....	97
16.2. Ejercicios propuestos	101
LECCIÓN 17: CONDICIONALES IV.....	103
17.1. Alternativa?.....	103
17.2. Ejercicios propuestos	106

LECCIÓN 18: LAS DIRECTIVAS.....	107
18.1. <i>#include</i>	107
18.2. <i>#define</i>	108
18.3. Ejercicios propuestos	111
LECCIÓN 19: CONDICIONALES V	113
19.1. Instrucción <i>switch</i>	113
19.2. Enunciado	113
19.3. Planteamiento de la solución.....	113
19.4. Código	114
19.5. Ejercicios propuestos	116
LECCIÓN 20: OTRO PROGRAMA COMPLETO	119
20.1. Versión completa.....	119
20.2. Ejercicios propuestos	124
LECCIÓN 21: EL CONCEPTO DE MENÚ	125
21.1. Definición.....	125
21.2. Enunciado	125
21.3. Construcción de las funciones	126
LECCIÓN DE EJERCICIOS.....	130
LECCIÓN 22: CICLOS I.....	135
22.1. Definición.....	135
22.2. Enunciado y algoritmo	135
22.3. Seudocódigo	136
22.4. Algoritmo técnico	138
LECCIÓN 23: CICLOS II.....	141
23.1. Prueba de escritorio.....	141
LECCIÓN 24: CICLOS III	147
24.1. Codificación usando sólo una función principal.....	147
24.2. Codificación usando funciones auxiliares	148
24.3. Ejercicios propuestos	150
LECCIÓN 25: CICLOS IV	151
25.1. Tipos de ciclos	151
25.2. Enunciado	151

25.3. Algoritmo conceptual.....	151
25.4. Algoritmo técnico	152
25.5. Prueba de escritorio.....	152
LECCIÓN 26: CICLOS V.....	157
26.1. Enunciado	157
26.2. Algoritmo técnico	157
26.3. Ciclo <i>for</i>	158
26.4. Ciclo <i>do-while</i>	159
26.5. Comparación entre estructuras cíclicas	159
26.6. Operadores.....	160
26.7. Ejercicios propuestos	160
LECCIÓN 27: VECTORES I.....	161
27.1. Aproximación	161
27.2. Concepto general	162
27.3. Algoritmos de solución	163
LECCIÓN 28: VECTORES II.....	167
28.1. Enunciado	167
28.2. Algoritmo conceptual.....	167
LECCIÓN 29: VECTORES III	171
29.1. Algoritmo completo.....	171
29.2. Prueba de escritorio.....	172
LECCIÓN 30: VECTORES IV	177
30.1. Algoritmo completo	177
30.2. Prueba de escritorio.....	178
LECCIÓN 31: VECTORES V	183
31.1. Algoritmo completo	183
31.2. Prueba de escritorio.....	184
LECCIÓN 32: VECTORES VI	189
32.1. Algoritmo completo.....	189
32.2. Código completo.....	190
32.3. Ejercicios propuestos	192

LECCIÓN: MATRICES I	195
33.1. Definición.....	195
33.2. Recorrido de la matriz.....	196
33.3. Algoritmo de recorrido	197
LECCIÓN 34: MATRICES II	201
34.1. Enunciado	201
34.2. Algoritmo conceptual.....	201
34.3. Algoritmo técnico	202
LECCIÓN 35: MATRICES III	207
35.1. Prueba de escritorio.....	207
35.2. Seguimiento	208
LECCIÓN 36: MATRICES IV	213
36.1. Prueba de escritorio.....	213
36.2. Seguimiento detallado	213
LECCIÓN 37: MATRICES V	217
37.1. Funciones	217
37.2. Código completo.....	219
37.3. Ejercicios propuestos	221
LECCIÓN 38: MATRICES VI	223
38.1. Enunciado	223
38.2. Algoritmo conceptual.....	223
38.3. Ejercicios propuestos	227
LECCIÓN 39: APUNTAORES I	229
39.1. Concepto general	229
39.2. Primera aproximación.....	230
39.3. Consideraciones	231
LECCIÓN 40: APUNTAORES II	235
40.1. Un ejemplo detallado.....	235
40.2. Despliegue de datos	236
LECCIÓN 41: APUNTAORES III	239
41.1. Variables globales.....	239
41.2. Variables locales	240

LECCIÓN 42: APUNTAORES IV	243
42.1. Paso de parámetros por valor.....	243
42.2. Paso de parámetros por referencia	245
LECCIÓN 43: APUNTAORES V	249
43.1. Niveles de direccionamiento.....	249
43.2. Consideraciones	251
LECCIÓN 44: APUNTAORES VI	253
44.1. Enunciado	253
44.2. Algoritmo conceptual.....	253
44.3. Código en C	254
44.4. Ejercicios propuestos	257
LECCIÓN 45: ARCHIVOS DE CARACTERES	259
45.1. Definición.....	259
45.2. Apuntadores tipo FILE	259
45.3. Ejercicios propuestos	263

INTRODUCCIÓN

¿Cómo enseñar programación de computadores? Esta ha sido una pregunta que, de manera recurrente, me he planteado a lo largo de los más de veinte años que tengo de experiencia docente. He recurrido a muchos libros que me guíen e indiquen la mejor forma de hacerlo. En ese camino, me he encontrado con muchos libros cuyo contenido versa sobre programación de computadores, pero ninguno de ellos me orientó hacia la forma de hacerlo.

Con el tiempo me arriesgué a pensar que la respuesta a mis necesidades podría estar en mí mismo y fue entonces cuando decidí estudiar un Doctorado en Ciencias de la Educación. Todas mis expectativas fueron colmadas mientras iba preparando una forma de enseñar programación de computadores, la gran respuesta que había inspirado no sólo mis estudios sino también mis inquietudes investigativas.

Este libro es una propuesta para enseñar programación de computadores basado en un lenguaje bastante popular conocido como el lenguaje C. Intento poner esta propuesta a consideración de docentes, estudiantes, profesionales y programadores en general que quieran aprender a programar y que encuentren la respuesta en sus páginas.

Este libro es producto de muchas investigaciones. Entre estas podemos contar el proceso que me llevó a concluir con éxito mi tesis doctoral que, precisamente, se fundamentó en el tema de la enseñanza en Ingeniería. Mi intención es que usted,

querido lector, aproveche estos contenidos al máximo, le saque todo el provecho posible y aprenda a programar bajo la combinación de dos paradigmas. Estos le ilustrarán cómo hacer que un computador haga por usted diferentes procesos aprovechando su alta velocidad de proceso. Reciba mi bienvenida a este libro y aprovéchelo al máximo.

Para mí, la respuesta ha sido encontrada a través de una dosificación del conocimiento en un formato de lecciones. Estas están pensadas para que usted las lea, realmente las lea y las disfrute pues, por ser tan cortas, podrá asimilarlas de una forma más simple. Espero que usted también encuentre en este libro esas respuestas que le permitan decir: “Aprendí a programar”.

Omar Iván Trejos Buriticá, PhD
omartrejos@utp.edu.co

¿Cómo usar este libro?

En primera instancia, le recomiendo que lo lea pausadamente. No se afane en avanzar, la apropiación y asimilación de los conceptos pertinentes a la programación de computadores implica tiempo pues no son exactamente los mismos conceptos que subyacen a la lógica natural y deliberativa que tenemos los seres humanos.

Lea y vuelva a leer, piense en lo que ha leído y, ante todo, realice (o, por lo menos, intente realizar) los ejercicios propuestos. Pregunte cuando tenga dudas y propóngase terminar los ejercicios siempre. No los deje a medio camino. Verá que cada vez que usted lleve un ejercicio hasta el final notará cómo su lógica humana se amplía para acudir a la lógica computacional en los casos donde corresponda.

Para el profesor

Utilice este libro al ritmo que sus estudiantes le permitan hacerlo, aprovechando que está diseñado en forma de lecciones. Cada lección proporciona el material para una sesión de clase. Usted y yo sabemos que lo más importante no es avanzar en un contenido, sino que los estudiantes realmente aprendan lo que se puede asimilar de ese eje temático. Sea muy paciente con la asimilación de la lógica de programación, de la programación y de las minucias de los lenguajes de programación. Estas tres instancias deben ser trasegadas por sus estudiantes para alcanzar el nivel de excelencia que usted ya ha adquirido en materia de programación.

Para el estudiante

Siga el ritmo que le indique el profesor. Revise los ejercicios resueltos, resuelva los ejercicios propuestos y, siempre sin excepción, pregunte. No se quede con dudas. Por simples que le parezcan, recuerde que cuando se trata de lógica, las dudas son también lógicas, pero si no se resuelven simplemente van quedando lagunas cuya resolución posteriormente puede llegar a ser más compleja. Comparta las soluciones encontradas con sus compañeros. Es muy enriquecedor alimentarse de la lógica de los demás y aportarles lo que, desde nuestra lógica, se pueda aportar. La forma excelsa para aprender a nadar es nadando. Asimismo, el camino óptimo para asimilar la lógica de programación es practicando, practicando y practicando.

PROGRAMACIÓN IMPERATIVA CON LENGUAJE C

La obra ofrece, a quienes se inician en la programación de computadores, una guía organizada desde la metodología hasta la teoría y práctica que requieren el conocimiento de matrices y funciones para resolver un problema de manera sencilla, explicando los elementos básicos que intervienen en la programación de computadores.

Los capítulos del libro siguen la línea invisible determinada por los conceptos propios de la programación de computadores, desde la forma de plantear una solución para un problema computable hasta el aprovechamiento de matrices y funciones para simplificarlo. Se explican conceptos teóricos al margen de cualquier lenguaje de programación, acudiendo tanto a conceptos comunes de estos como a los diferentes paradigmas de la programación tradicional y moderna.

El libro está dirigido a estudiantes y docentes de las áreas de Informática, Programación, Ingeniería de Sistemas, y en general a todos los interesados en la programación de computadores.

Colección: Ingeniería y salud en el trabajo
Área: Informática

ECOE
EDICIONES

www.ecoeediciones.com

Incluye

- ▶ Ejemplos explicados detalladamente a la luz de cada concepto estudiado.
- ▶ Ejercicios propuestos al finalizar cada capítulo, con diferentes niveles de exigencia.
- ▶ Conceptos comunes a los paradigmas de la programación moderna.

Omar Iván Trejos Buriticá

Ingeniero de sistemas de la Universidad INCCA de Colombia. Especialista en Instrumentación física y Master en Comunicación educativa de la Universidad Tecnológica de Pereira. PhD en Ciencias de la educación (RUDECOLOMBIA – UTP). Ha sido docente, desde 1996, de la Facultad de Ingenierías, Ingeniería de Sistemas y Computación de la Universidad Tecnológica de Pereira, donde se ha desempeñado, además, como Decano de la Facultad y Director del programa de Ingeniería de Sistemas y Computación.

ISBN 978-958-771-543-9



9 789587 715439

e-ISBN 978-958-771-544-6