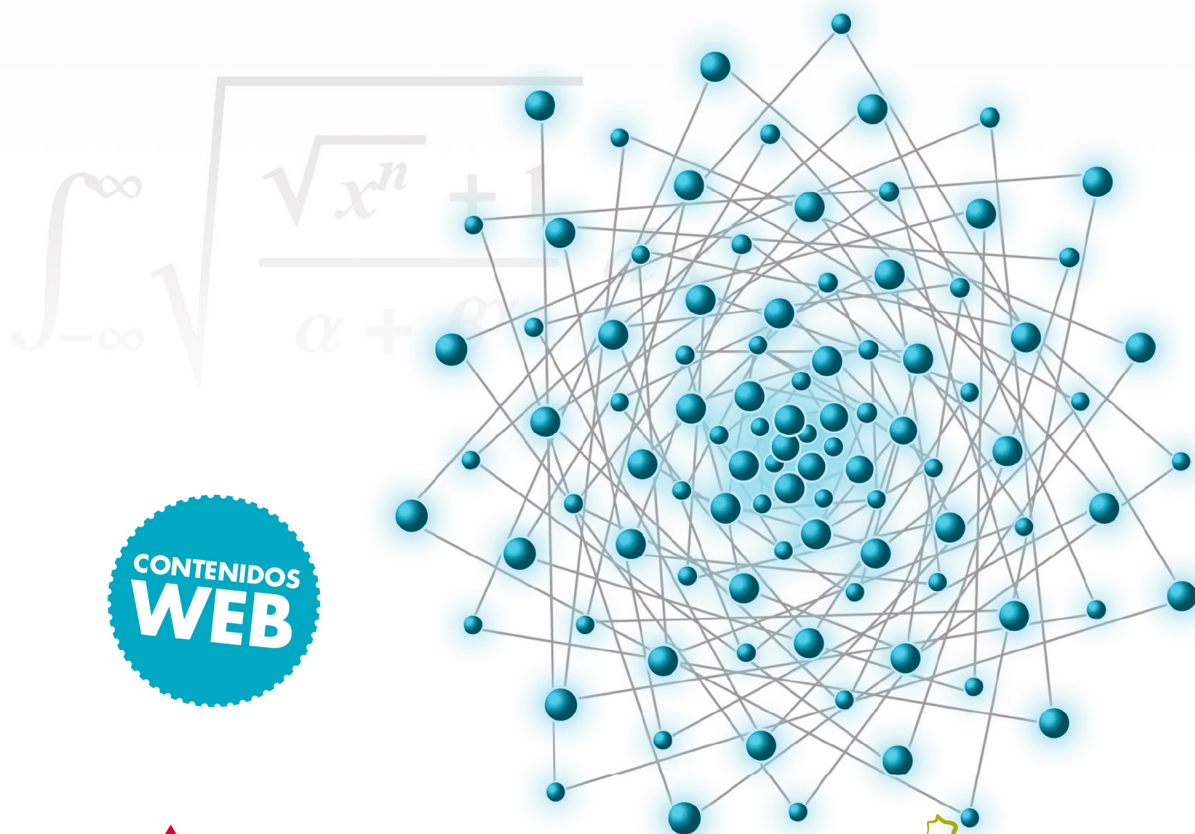


OFELIA D. CERVANTES VILLAGÓMEZ • DAVID BÁEZ LÓPEZ
ANTONIO ARÍZAGA SILVA • ESTEBAN CASTILLO JUÁREZ

pythonTM



CON **APLICACIONES** A LAS MATEMÁTICAS, INGENIERÍA Y FINANZAS



CONTENIDOS
WEB

 **Alfaomega**

 **Marcombo**

Python con aplicaciones a las matemáticas, ingeniería y finanzas

Ofelia Delfina Cervantes Villa Gómez

José Miguel David Báez López

Antonio Arízaga Silva

Esteban Castillo Juárez

Acceda a www.marcombo.info
para descargar gratis
contenidos adicionales
complemento imprescindible de este libro

Código:

PYTHON2

Python con aplicaciones a las matemáticas, ingeniería y finanzas

Ofelia Delfina Cervantes Villa Gómez
José Miguel David Báez López
Antonio Arízaga Silva
Esteban Castillo Juárez



Python con aplicaciones a las matemáticas, ingeniería y finanzas
Ofelia Delfina Cervantes Villa Gómez, José Miguel David Báez López,
Antonio Arizaga Silva, Esteban Castillo Juárez

Derechos reservados © Alfaomega Grupo Editor, S.A. de C.V., México
Primera edición: agosto 2017
ISBN: 978-607-622-673-5

Primera edición: MARCOMBO, S.A. 2018

© 2018 MARCOMBO, S.A.
www.marcombo.com

«Cualquier forma de reproducción, distribución, comunicación pública o transformación de esta obra sólo puede ser realizada con la autorización de sus titulares, salvo excepción prevista por la ley. Diríjase a CEDRO (Centro Español de Derechos Reprográficos, www.cedro.org) si necesita fotocopiar o escanear algún fragmento de esta obra».

ISBN: 978-84-267-2587-5
D.L.: B-26659-2017

Impreso en Servicepoint
Printed in Spain

Acerca de los autores



Ofelia Cervantes Villagómez Ingeniera en Sistemas Computacionales (UDLAP), DEA en Informática (EN-SIMAG, Francia) y Doctorado en Computación (INPG, Francia). Es profesora del Departamento de Computación, Electrónica y Mecatrónica en la UDLAP desde 1988 donde fue Jefa del Departamento de 1990 a 1993. Sus áreas de especialidad son la inteligencia artificial, bases de datos y sistemas distribuidos.

Fue Decana de la Escuela de Ingeniería y posteriormente Decana de Asuntos Internacionales de 1994 a 2005.

De 2006 a 2008 creó y dirigió la Escuela de Verano de la UDLAP. De 2011 a 2012 fue fundadora y primera Directora del Programa de Liderazgo para Jóvenes Indígenas. Ha colaborado durante décadas en la vinculación de la comunidad académica especializada en TICs, con sus homólogos en EUA, Francia y Alemania. Ha publicado en congresos y revistas nacionales e internacionales. Es coautora de libros y capítulos de libros en su área de especialidad.

Desde 2003 es Cónsul Honoraria de Francia en Puebla. En Abril 2014 fue condecorada por el Presidente de la República Francesa como Caballero de la Orden Nacional del Mérito como reconocimiento a los servicios brindados a Francia en la protección consular y promoviendo la cooperación académica con México. Es miembro de la Academia Mexicana de Computación.

David Báez López. Licenciado en Física (UAP), Maestro en Ciencias (Universidad de Arizona), Doctor en Ingeniería Eléctrica (Universidad de Arizona). De 1979 a 1985 fue investigador del Instituto Nacional de Astrofísica, Óptica y Electrónica en Tonantzintla, Puebla, siendo Coordinador de Electrónica y Director Técnico Interino. De 1985 a 2015 fue profesor del Departamento de Computación, Electrónica y Mecatrónica de la Universidad de las Américas Puebla (UDLAP) en Cholula, Puebla, México. En la UDLAP fungió como jefe del Departamento de Ingeniería Electrónica y como Presidente del Comité de Evaluación de la Facultad. Así mismo, es fundador del Congreso Internacional de Electrónica, Comunicaciones y Computadoras que en 2017 está en su 27^a edición y que se lleva a cabo en la UDLAP.



Ha publicado más de cien papers en revistas y congresos nacionales e internacionales, respectivamente, es autor y coautor de capítulos de libros y de trece libros acerca de simulación de circuitos, MATLAB, y procesamiento analógico de señales. Como investigador fue miembro del Sistema Nacional de Investigadores de 1984 hasta 2014.



Juan Antonio Arízaga Silva recibió el grado de Maestro en Ciencias de la Universidad de las Américas Puebla en 2006 y el título de licenciado en electrónica en 2003 por la Benemérita Universidad Autónoma de Puebla. Actualmente es profesor investigador de tiempo completo en la Universidad Politécnica de Puebla en el área de Sistemas Automotrices. Ha escrito más de 20 artículos técnicos en diversas revistas nacionales e internacionales. Las áreas de interés de sus investigaciones incluyen sistemas embebidos, sistemas de telemetría automotriz y sistemas electrónicos de potencia.

Esteban Castillo Juárez es Licenciado en ciencias de la computación (2009) y maestro en ciencias de la computación (2012) por parte de la Benemérita Universidad Autónoma de Puebla. Actualmente es estudiante de doctorado en ciencias de la computación en la Universidad de las Américas Puebla. Sus áreas de interés incluyen el procesamiento de lenguaje natural, minería de datos, aprendizaje automático, análisis de redes sociales y teoría de grafos.

Ha publicado mas de 15 artículos internacionales relacionados con el tratamiento automático del texto utilizando representaciones basadas en grafos con el lenguaje de programación Python, con el cual ha trabajado activamente por más de 5 años.



Con gran cariño a mi hijo Rémy, estrella brillante en mi vida, razón profunda para cada día ser mejor persona.

OCV

Este esfuerzo está dedicado a mis hijos Laura Michele y David Alfredo que han sido la luz en mi camino.

DBL

Con mucho cariño para mi familia y mi asesora de tesis Dra. Ofelia Cervantes.

ECJ

A mis hijos, mi esfuerzo presente y futuro siempre será por ustedes. A Gaby, Dios nos puso en el mismo camino, vayamos tomados de la mano.

AAS

Agradecimientos

El presente libro fue posible gracias al apoyo de muchas personas que con su valiosa ayuda nos permitieron que se volviera una realidad.

Este trabajo se enriqueció con la participación de muchos de nuestros estudiantes de la Universidad de las Américas Puebla, a los que les enseñamos este material en su versión preliminar. Gracias a ellos, y a su gran entusiasmo y curiosidad, pudimos mejorar la explicación y los ejemplos ilustrativos del material. Un especial agradecimiento a nuestros alumnos que anonimamente checaron los programas presentados en este libro.

Un especial agradecimiento para nuestras familias por el tiempo que no estuvimos con ellas para poder terminar este proyecto.

Finalmente, agradecemos al Dr. Vicente Alarcón Aquino, Director Académico del Departamento de Computación, Electrónica y Mecatrónica de la Universidad de las Américas Puebla, por el apoyo dado a la realización de este proyecto.

Mensaje del Editor

Una de las convicciones fundamentales de Alfaomega es que los conocimientos son esenciales en el desempeño profesional, ya que sin ellos es imposible adquirir las habilidades para competir laboralmente. El avance de la ciencia y de la tecnología hace necesario actualizar continuamente esos conocimientos y, de acuerdo con esta circunstancia, Alfaomega publica obras actualizadas, con alto rigor científico y técnico, y escritas por los especialistas del área respectiva más destacados.

Consciente del alto nivel competitivo que debe de adquirir el estudiante durante su formación profesional, Alfaomega aporta un fondo editorial que se destaca por sus lineamientos pedagógicos que coadyuvan a desarrollar las competencias requeridas en cada profesión específica.

De acuerdo con esta misión, con el fin de facilitar la comprensión y apropiación del contenido de esta obra, cada capítulo inicia con el planteamiento de los objetivos del mismo y con una introducción en la que se plantean los antecedentes y una descripción de la estructura lógica de los temas expuestos. Asimismo a lo largo de la exposición, se presentan ejemplos desarrollados con todo detalle y cada capítulo concluye con un resumen y una serie de ejercicios propuestos.

Además de la estructura pedagógica con que está diseñado el contenido de nuestros libros, Alfaomega hace uso de los medios impresos tradicionales en combinación con las Tecnologías de la Información y las Comunicaciones (TIC) para facilitar el aprendizaje. Correspondiente a este concepto de edición, todas nuestras obras tienen su complemento en una página Web en donde el alumno y el profesor encontrarán lecturas complementarias así como programas desarrollados en relación con temas específicos de la obra.

Los libros de Alfaomega están diseñados para ser utilizados en los procesos de enseñanza aprendizaje, y pueden ser usados como textos en diversos cursos o como apoyo para reforzar el desarrollo profesional. De esta forma Alfaomega espera contribuir así a la formación y al desarrollo de profesionales exitosos para beneficio de la sociedad, y espera ser su compañera profesional en este viaje de por vida por el mundo del conocimiento.



Contenido

Prólogo	XVII
Plataforma de contenidos interactivos	XIX
1. Introducción	1
1.1. Introducción	2
1.2. Evolución de la computadora	2
1.3. Arquitectura de una computadora	6
1.3.1. Instrucciones y datos de una computadora	7
1.3.2. Sistemas numéricos decimal y binario	7
1.3.3. Otros sistemas numéricos	11
1.3.4. Estructura de la memoria	12
1.3.5. Lenguajes de una computadora	13
1.4. El lenguaje Python	14
1.4.1. Descarga de Python	15
1.4.2. La interfase de Python	15
1.5. Organización del libro	16
2. Fundamentos de algoritmos y de Python	17
2.1. Introducción	18
2.2. ¿Qué es un algoritmo?	18
2.3. Pseudocódigo	21
2.4. Variables	22
2.4.1. Tipos de variables	22
2.4.2. Asignación de valores a variables	23
2.4.3. Inicialización de variables	23
2.4.4. Operaciones básicas	24
2.4.5. Operaciones con variables enteras	25
2.4.6. Operaciones con enteros y reales	25

2.5. Partes de un algoritmo	26
2.6. Algoritmos en pseudocódigo	27
2.7. Lenguaje Python	29
2.7.1. El ambiente de Python	30
2.8. Estructura de un algoritmo en Python	32
2.9. Variables	32
2.10. Bibliotecas y encabezados	34
2.11. Operadores	35
2.11.1. Operadores aritméticos	35
2.11.2. Operadores relacionales	36
2.11.3. Operadores lógicos	36
2.11.4. Otras operaciones booleanas	38
2.11.5. Operadores de asignación	39
2.12. Comentarios	39
2.13. Entrada y salida de datos en Python	40
2.13.1. Despliegue de datos	40
2.13.2. Entrada de datos	42
2.13.3. Tabulador	43
2.14. Algoritmos sencillos	43
2.15. Variables alfanuméricas en Python	48
2.15.1. Operaciones con cadenas	49
2.15.2. Concatenación o suma de cadenas	50
2.15.3. Multiplicación de cadenas	50
2.16. Listas	52
2.16.1. Diccionarios	53
2.17. Instrucciones de Python del Capítulo 2	54
2.18. Conclusiones	54
2.19. Ejercicios	55
3. Condiciones	59
3.1. Introducción	60
3.2. Condiciones	61
3.2.1. La condición en Python	63
3.2.2. Ejercicios de condiciones simples	68
3.3. La condición Si - Si_no	73
3.3.1. Condición if - else	73

3.3.2. Ejemplos adicionales de condiciones if-else	75
3.4. Condiciones anidadas	77
3.4.1. if anidado en Python	78
3.4.2. Ejemplos adicionales con condiciones anidadas	80
3.5. Casos	83
3.6. Instrucciones de Python del Capítulo 3	89
3.7. Conclusiones	89
3.8. Ejercicios	89
4. Ciclos en Python	95
4.1. Introducción	96
4.2. Ciclos Mientras	96
4.2.1. El ciclo Mientras en Python	100
4.3. Ciclos Para	106
4.3.1. La función range	112
4.4. Ciclos anidados	113
4.5. La instrucción continue	123
4.6. La instrucción break	124
4.7. Ejemplos adicionales	126
4.8. Instrucciones de Python del Capítulo 4	132
4.9. Conclusiones	133
4.10. Ejercicios	133
5. Cadenas, Listas, Diccionarios y Tuplas	137
5.1. Introducción	138
5.2. Cadenas	138
5.2.1. Longitud de una cadena	141
5.2.2. Separación de una cadena	141
5.2.3. Operaciones con cadenas	143
5.2.4. Concatenación o suma de cadenas	143
5.2.5. Multiplicación de cadenas	144
5.2.6. Inmutabilidad de las cadenas	144
5.2.7. Otras operaciones con cadenas	145
5.2.8. Condiciones	148
5.2.9. Ciclos	149
5.3. Listas	151

5.4.	Definición de listas	152
5.4.1.	Operaciones con listas	152
5.4.2.	Mutabilidad	154
5.4.3.	Conversión de lista de cadenas a cadena	155
5.4.4.	Otras operaciones con listas	156
5.5.	Tuplas	159
5.5.1.	Intercambio de valores	159
5.5.2.	Operaciones con tuplas	160
5.6.	Diccionarios	161
5.6.1.	Otras operaciones para diccionarios	162
5.7.	Instrucciones de Python del Capítulo 5	164
5.8.	Conclusiones	164
5.9.	Ejercicios	164
6.	Arreglos I: Vectores	169
6.1.	Introducción	170
6.2.	Introducción a arreglos	170
6.3.	Vectores	171
6.3.1.	Acceso a vectores	175
6.3.2.	Importancia del tamaño de un vector	179
6.4.	Vectores en Python	180
6.4.1.	Vectores por comprensión	181
6.5.	Ejemplos con vectores en Python	182
6.5.1.	Importancia del tamaño de un vector en Python	193
6.5.2.	Inicialización de los vectores	194
6.5.3.	La instrucción append	194
6.6.	Ordenamiento de vectores	196
6.6.1.	Ordenamiento por selección	196
6.6.2.	Ordenamiento de burbuja	200
6.7.	Búsquedas	204
6.7.1.	Búsqueda binaria	205
6.8.	Instrucciones de Python del Capítulo 6	208
6.9.	Conclusiones	208
6.10.	Ejercicios	208

7. Arreglos II: Matrices	211
7.1. Introducción	212
7.2. Matrices	212
7.3. Arreglos en Python	214
7.3.1. Generación de arreglos por indexación	215
7.3.2. Generación de arreglos por comprensión	217
7.4. Métodos alternos de escritura de matrices	218
7.5. Selección de filas y columnas de un arreglo	220
7.5.1. Filas de un arreglo	220
7.5.2. Columnas de un arreglo	221
7.6. Suma, resta y multiplicación de matrices	221
7.6.1. Suma y resta de matrices	222
7.6.2. Suma y resta de matrices en Python	223
7.6.3. Multiplicación de matriz por un escalar	225
7.6.4. Multiplicación de matriz por una matriz	227
7.6.5. Producto de matrices en Python	229
7.7. Matrices especiales	231
7.7.1. La matriz identidad	231
7.7.2. La matriz identidad en Python	231
7.7.3. La matriz transpuesta	233
7.7.4. Transpuesta de un arreglo en Python	234
7.7.5. Obtención de la transpuesta por comprensión	236
7.7.6. La matriz simétrica	237
7.8. Ejemplos	239
7.9. Conclusiones	245
7.10. Ejercicios	245
8. Subalgoritmos	249
8.1. Introducción	250
8.2. Subalgoritmos	250
8.3. Funciones	252
8.4. Funciones en Python	255
8.4.1. Funciones de Python	256
8.4.2. Funciones definidas por el usuario	256
8.5. Procedimientos	261
8.5.1. Recursividad	266

8.6. Funciones lambda	267
8.7. Llamado por valor y llamado por referencia	268
8.8. Variables locales y globales	269
8.9. Ejemplos adicionales	272
8.10. Instrucciones de Python del Capítulo 8	292
8.11. Conclusiones	292
8.12. Ejercicios	292
9. Entrada y salida y de datos con archivos	295
9.1. Introducción	296
9.2. Escritura de datos en un archivo	296
9.2.1. Escritura de datos alfanuméricos	297
9.2.2. La instrucción <code>with</code>	300
9.3. Escritura de datos numéricos	300
9.4. Lectura de datos de un archivo	303
9.4.1. Lectura de datos de un archivo	304
9.5. Lectura y escritura de datos en Excel	308
9.6. Instrucciones de Python del Capítulo 9	313
9.7. Conclusiones	314
9.8. Ejercicios	314
10. Programación orientada a objetos	315
10.1. Introducción	316
10.2. Conceptos asociados a la POO	316
10.3. Primera clase en Python	319
10.4. Creación de la clase <code>NumeroComplejo</code>	321
10.5. Declaración y uso de <code>Setters</code> y <code>Getters</code>	322
10.6. Sobreescritura de operadores	324
10.7. Herencia	326
10.8. Sobreescritura de métodos	330
10.9. Ejemplos	331
10.9.1. Creación y movimientos de una cuenta bancaria	331
10.9.2. Cálculo de la media aritmética y la desviación estándar	334
10.9.3. Operaciones con matrices	337
10.9.4. Figuras geométricas	341
10.9.5. Lista enteros	344

10.10. Instrucciones de Python del Capítulo 10	349
10.11. Conclusiones	350
10.12. Ejercicios	350
11. Graficación en Python	351
11.1. Introducción	352
11.2. Visualización de datos	352
11.3. Gráficas en 2 dimensiones	353
11.4. Figuras múltiples	360
11.5. Subgráficas	362
11.6. Otros tipos de gráficas bidimensionales	365
11.6.1. Gráfica polar	365
11.6.2. Gráfica de pie	366
11.6.3. Gráfica de histograma	367
11.6.4. Gráfica de <code>stem</code> o de puntos	368
11.7. Opciones de gráficas	370
11.8. Gráficas tridimensionales	371
11.9. Instrucciones de Python del Capítulo 11	379
11.10. Conclusiones	380
11.11. Ejercicios	381
12. Geolocalización y Análisis de Sentimientos	383
12.1. Geolocalización	384
12.2. El módulo <code>geopy</code>	384
12.2.1. Geolocalización de un punto de interés	385
12.2.2. Distancia de dos puntos de interés	387
12.2.3. Visualización de distintos puntos de interés	388
12.3. Análisis de sentimientos de Twitter	390
12.3.1. Extracción de tweets	391
12.3.2. La base de datos MongoDB	395
12.3.3. Análisis de los tweets	398
12.4. Conclusiones	403
A. Instalación y configuración	405
A.1. Introducción	406
A.2. Instalación de Python	406

A.3. Instalación de easy_install y pip	408
A.4. Instalación de Numpy	411
A.5. Instalación de Scipy	412
A.6. Instalación de Matplotlib	413
A.7. Instalación de Tweepy	415
A.8. Instalación de Pymongo	416
A.9. Instalación de Geopy	417
A.10. Instalación de Matplotlib Basemap	418
B. Creación de ejecutables en Python	421
B.1. Introducción	422
B.2. Instalación de pyinstaller	422
B.3. Creación de ejecutables	422
B.4. Ejemplo	423

Prólogo

Cuando nos propusimos escribir un libro sobre programación para alumnos de primer semestre de licenciatura, examinamos las tendencias en el uso de lenguajes de programación en la industria y en las universidades. El resultado fue que Python era el lenguaje que estaba tomando fuerza y colocándose en los primeros lugares en la lista de los lenguajes más usados. Hay varias razones para este hecho. Python es software libre y se cuenta con una gran cantidad de módulos para realizar cómputo científico que encuentra aplicaciones en muchas áreas del conocimiento científico, desde medicina, ingeniería, ciencias, finanzas, economía, por mencionar algunas cuantas. Además, Python se ha erigido como un rival de respeto de MATLAB® con la ventaja de tener una distribución gratuita. Al momento de la publicación de este libro, Python se encuentra posicionado en el cuarto lugar de la lista de lenguajes de programación más usados por programadores, pero colocado en primer lugar entre los lenguajes de programación de muy alto nivel. Adicionalmente, el IEEE lo ha colocado en tercer lugar entre los lenguajes de programación. Estas estadísticas nos hicieron seleccionar a Python como el lenguaje de programación para este libro.

El contenido del presente libro ha sido parte del curso introductorio de programación para estudiantes de ciencias, ingeniería y animación digital a nivel licenciatura de la Universidad de las Américas Puebla. Constituye para muchos estudiantes el primer contacto con un lenguaje de programación. Durante el libro hacemos énfasis en la práctica de las instrucciones, inicialmente en algoritmos sencillos y aumentando el nivel de complejidad de acuerdo al avance de los temas. El presente material consiste en una versión de los temas que hemos impartido durante los últimos dos años. La importancia del lenguaje de programación Python se hace patente dado el crecimiento en su uso en las universidades y en la industria, a un grado tal que las nanocomputadora Raspberry Pi lo usan como lenguaje de programación.

Un aspecto importante del libro consiste en desarrollar desde los primeros capítulos la descripción, análisis y desarrollo de los algoritmos que posteriormente se implementan en el lenguaje de programación Python. De esta manera el usuario va de una descripción del problema en el idioma español, pero con una estructura definida, a una implementación formal en un lenguaje de programación. De esta manera el lector ve como se va construyendo un algoritmo para de una manera sencilla poder implementarlo en Python. Con este propósito el libro desarrolla más de 150 ejemplos, con los cuales el lector aprende los conceptos presentados en cada capítulo.

Como en toda rama de la ingeniería, en la programación con Python, es necesario tener un concepto integral de aprendizaje que complemente la teoría y los ejemplos resueltos. Por lo tanto, cada capítulo tiene como complemento una serie de ejercicios que el lector puede resolver y que le permiten ganar más experiencia en la solución de problemas implementando sus algoritmos en lenguaje Python.

Tenemos confianza en que el libro permitirá a todos sus lectores aprender a diseñar algoritmos implementándolos en el lenguaje Python.

Competencias Generales

Con el material de este libro el lector desarrollará las siguientes competencias:

1. Desarrollar algoritmos en pseudocódigo.
2. Implementar los algoritmos en lenguaje Python.
3. Graficar resultados de los algoritmos.
4. Usar Python para distintas aplicaciones como geolocalización, análisis de sentimientos, álgebra lineal, cadenas de Markov, procesamiento de señales, operaciones en finanzas, entre otras.

Al final del libro se incluyen tres Apéndices. El primer Apéndice explica como instalar módulos adicionales para realizar otras funciones en Python tales como graficar, realizar otras operaciones numéricas, financieras, geolocalización, entre otras. El segundo explica como realizar archivos ejecutables en Python, de tal manera que no se requiera tener instalado Python para su ejecución. El tercer y último apéndice señala las principales diferencias entre Python 2 y Python 3.

Se espera que el lector aprenda a desarrollar algoritmos en pseudocódigo, los cuales se puedan traducir al lenguaje de programación Python o a cualquier otro lenguaje que el lector requiera implementar.

Plataforma de contenidos interactivos

Para tener acceso al material de la plataforma de contenidos interactivos de este libro, siga los pasos de la primera página del libro.

NOTA: Se recomienda respaldar los archivos descargados de la página web en un soporte físico.



Capítulo 1

Introducción

- 1.1 Introducción**
- 1.2 Evolución de la computadora**
- 1.3 Arquitectura de una computadora**
- 1.4 El lenguaje de Python**
- 1.5 Organización del libro**

Al igual que los hijos biológicos de generaciones anteriores, las máquinas representan la mejor esperanza de la humanidad para un futuro a largo plazo.

Moravec

Objetivos

En este capítulo aprenderemos los conceptos básicos de la arquitectura de una computadora típica y de cómo se controlan los datos y se usan las instrucciones para realizar las aplicaciones diseñadas por los usuarios.

1.1 Introducción

Los primeros años del siglo 21 han visto un crecimiento sorprendente en el uso y creación de nuevas computadoras. Esto requiere una gran cantidad de expertos que sepan afrontar los retos de dicho crecimiento. Estos expertos deben conocer entre otras cosas lenguajes de programación modernos que permitan que una computadora ejecute aplicaciones, que tenga acceso a Internet, que realice cálculos matemáticos complicados, que comunique con colegas y amigos en cualquier parte del mundo, entre otras actividades que realiza un usuario con una computadora. Este libro aborda el lenguaje de programación Python a nivel introductorio. Sin embargo, el lector será capaz de realizar gráficas, decisiones, manipular archivos y ver como Python se puede usar en distintas aplicaciones.

1.2 Evolución de la computadora

La historia de la computación se remonta muchos siglos atrás. El instrumento de cálculo más antiguo que se conoce es el ábaco el cual fue usado en la antigua China. En el siglo 17 Pascal y Leibnitz desarrollaron calculadoras mecánicas que son las antecesoras de las calculadoras electrónicas de nuestros días.

El primer dispositivo que se considera una computadora fue propuesto por Charles Babbage en 1830; véase la figura 1.1. Un excéntrico matemático inglés, Babbage deseaba automatizar el cálculo de tablas matemáticas. Para este propósito diseñó la Máquina de Diferencias. Aunque nunca la pudo construir totalmente, la Máquina de

Diferencias contenía las partes fundamentales de una computadora moderna, como son la memoria para datos e instrucciones, una unidad de entrada/salida y la posibilidad de realizar decisiones en la secuencia de instrucciones.

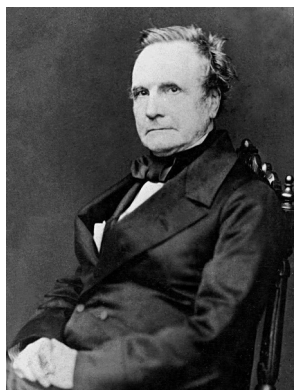


Figura 1.1 Charles Babbage.

Con el paso del tiempo surgieron las grandes compañías como IBM, entre otras, que se dedicaban a diseñar, producir y vender equipo de cómputo usando dispositivos electromecánicos. Fundada en 1911 con el nombre de Computing Tabulating Recording Corporation que se cambió luego a International Business Machines, IBM, se dedicaba a realizar calculadoras electromecánicas.

Es hasta 1937 cuando Howard Aiken propuso la Calculadora Automática Controlada Secuencialmente (Automatic Sequence Controlled Calculator). La construcción de esta máquina se inició en 1939 y se terminó en 1944 y se conoció como Mark I.

Esta máquina era principalmente electromecánica y se usaba para cálculos de trayectorias de misiles por el ejército de los Estados Unidos. Para acelerar los cálculos, la Mark I tenía varios procesadores en paralelo controlados por una unidad de control.

En la misma época en la Gran Bretaña se daba el desarrollo de la computadora Colossus bajo la guía de Alan Turing. Esta máquina contribuyó a la decodificación de la máquina Enigma la cual era usada por los alemanes durante la Segunda Guerra Mundial para codificar sus mensajes. Desafortunadamente, el gobierno de la Gran Bretaña ordenó su destrucción al final de la guerra.

Mientras tanto, los usuarios de Mark I se dieron cuenta de la necesidad de una máquina más poderosa y rápida por lo que el ejército de los Estados Unidos contrató con la Universidad de Pennsylvania para desarrollar la computadora ENIAC, considerada como la primera computadora electrónica ya que estaba formada por bulbos electrónicos principalmente. El asesor matemático de este proyecto fue John von Neumann a quien muchos investigadores de la computación consideran el más importante personaje en la historia de la computación.

Aunque ENIAC fue básicamente una versión electrónica de la Mark I, la rapidez de la electrónica eliminó la necesidad de los procesadores en paralelo¹, además almacenaba en memoria la secuencia de instrucciones.

Muchos desarrollos siguieron a estas máquinas pioneras de la computación, pero las bases de las computadoras modernas estaban ya trazadas. Compañías como IBM,

¹Las computadoras actuales realizan cálculos en paralelo para aumentar la velocidad de cómputo.

Digital, Sperry Rand, Control Data Corporation (CDC), Data General, Apple, HP, se erigieron como fabricantes de computadoras, cada vez más rápidas, poderosas y compactas.

Las computadoras de aquellos años pertenecían a lo que se conoce como **Main Frames**. Dicho término se aplica a aquellas computadoras que dan servicio a grandes corporativos, oficinas gubernamentales, bancos, entre otros. Algunos fabricantes de este tipo de computadoras fueron IBM y Control Data Corporation.

En 1960 se creó la primera minicomputadora. Se consideraba que una minicomputadora era una computadora que tenía un precio menor a \$25,000.00 dólares. Es la compañía Digital Equipment Corporation (DEC) que produjo la primera minicomputadora PDP-8, a la que le siguieron otras como la PDP-11 y la VAX1130. Otras compañías también produjeron minicomputadoras como es el caso de Data General y Wang Laboratories.

En la década de los 1970 se presentaron las primeras redes de computadoras en Estados Unidos y en Europa.

Es en la década de 1970 cuando empezaron a surgir las computadoras de escritorio que se podían conectar a un televisor para que funcionara como monitor. Estas computadoras tenían una memoria limitada. Las compañías más conocidas eran Tandy Corp. y Commodore. En esa misma década nació Apple Computer Inc., fabricante de las computadoras Macintosh.

Los últimos años han visto un aumento en la velocidad de las computadora así como una reducción en el costo y tamaño. Además, el Internet se ha puesto al servicio del público en general a través de tabletas y teléfonos inteligentes que hacen que las computadoras estén presentes en todos los aspectos de la vida moderna a tal grado que es imposible imaginarnos la vida sin computadoras y sin Internet.

Tabla 1.1 Crónica de la computación

Año	Nombre	Hecha por	Comentarios
1830	Máquina Analítica	Babbage	Primer intento de construir una computadora digital.
1936	Z1	Zuse	Primera máquina calculadora con relevadores.
1943	Colossus	Gobierno Británico	Primera computadora electrónica.
1944	Mark I	Aiken	Primera computadora de propósito general.
1946	ENIAC I	Eckert/Mauchley	Primera computadora de la historia moderna.
1949	EDSAC	Wilkes	Primer programa almacenado en la computadora.

Continúa

Tabla 1.1 Continuación

Año	Nombre	Hecha por	Comentarios
1951	Whirlwind I	M.I.T	Primer programa en tiempo real.
1952	IAS	Von Neumann	La mayoría de las computadoras usan este diseño.
1960	PDP-1	DEC	Primera minicomputadora (se vendieron 50).
1961	1401	IBM	Máquina para negocios muy popular.
1962	7094	IBM	Dominó el cómputo científico en los 60's.
1963	B5000	Burroughs	Primera máquina en usar lenguaje de alto nivel.
1964	360	IBM	Primera familia de computadoras.
1964	6600	CDC	Primera supercomputadora.
1965	PDP-8	DEC	Primera minicomputadora popular.
1970	PDP-11	DEC	Minicomputadora más vendida en los 70s.
1974	8080	Intel	Primera computadora de 8 bits en un circuito integrado.
1974	CRAY-1	Cray	Primera supercomputadora.
1975	Microsoft Corp.		Bill Gates y Paul Allen fundan Microsoft Corp.
1976	Apple Computer Inc.		Steve Jobs y Steve Wozniak son los fundadores.
1978	VAX	DEC	Superminicomputadora de 32 bits.
1981	IBM PC	IBM	Empieza la era de la computadora personal (PC).
1981	Osborne-1	Osborne	Primera computadora portátil.
1983	Lisa	Apple	Primera computadora personal con interfase gráfica y mouse.
1984	Macintosh	Apple	Primera computadora para todo público.
1985	386	Intel	Primer circuito integrado de 32 bits.
1985	MIPS	MIPS	Primera máquina RISC.
1987	SPARC	Sun	Primera estación de trabajo.
1990	WWW	Tim Berners-Lee	En el CERN de Ginebra se crea la World Wide Web.
1991	Python	Guido von Rossum	Lanzamiento del lenguaje.
1992	Alpha	DEC	Primera computadora personal de 16 bits.

Continúa

Tabla 1.1 Continuación

Año	Nombre	Hecha por	Comentarios
1993	Newton	Apple	Primera computadora del tipo Palmtop.
2001	PowerBook G4	Apple	Línea de notebooks de Apple.
2010	iPad	Apple	Hace el acceso a Internet y Apps muy sencillo.

1.3 Arquitectura de una computadora

Una computadora está formada por 5 partes principales. Estas partes son:

- Unidad de memoria. Aquí se guardan los datos y las instrucciones. Típicamente es un disco duro, una memoria RAM, o una unidad externa como un disco duro o una memoria USB.
- Unidad de entrada. La parte por donde se reciben los datos y las instrucciones de la aplicación. Puede ser un teclado o un archivo de datos o de instrucciones.
- Unidad de salida. Es la sección por donde se despliegan los resultados de la aplicación. Puede ser una pantalla, un listado, una hoja de cálculo o un archivo de datos.
- Unidad Lógica Aritmética (Arithmetic Logic Unit, ALU). Es la parte encargada de realizar las operaciones aritméticas, lógicas y de decisión de acuerdo al diseño de la aplicación. Junto con la unidad de control forma la Unidad de Procesamiento Central o CPU (Central Processor Unit).
- Unidad de control. Es la parte encargada de decodificar las instrucciones y realizar las operaciones que se hayan programado en la aplicación. Junto con la Unidad Lógica Aritmética forman el CPU (central processor unit).

La manera en cómo están interconectadas estas partes se conoce como la arquitectura de la computadora. Una arquitectura básica típica se muestra en la figura 1.2. Las diferentes unidades están interconectadas por el **bus de datos** y por el **bus de instrucciones**. Por el bus de datos se transmiten los datos entre las distintas unidades de la computadora. Por el bus de instrucciones se transmiten las instrucciones que se decodifican para efectuar las operaciones programadas en la aplicación.

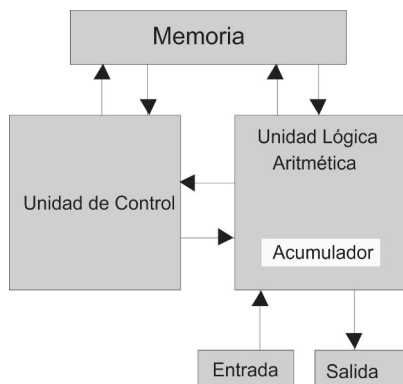


Figura 1.2 Arquitectura básica de una computadora.

1.3.1 Instrucciones y datos de una computadora

Una computadora necesita de instrucciones para realizar una tarea dada por el usuario. Estas instrucciones, así como los datos, están escritas en una notación binaria. De esta manera, una computadora realiza todas las operaciones necesarias usando el sistema numérico binario o de base 2. La siguiente sección explica en detalle este sistema numérico.

1.3.2 Sistemas numéricos decimal y binario

Para explicar el sistema binario primero explicamos el sistema numérico decimal o de base 10. Consideremos un número como el 2943. En este número tenemos cuatro dígitos decimales. El de mayor valor es el 2 que corresponde a los millares o 10^3 . Sigue el nueve que corresponde a las centenas o 10^2 . El 4 corresponde a las decenas o $10 = 10^1$. Finalmente el 3 corresponde a las unidades $1 = 10^0$. Cada uno de estos dígitos decimales ocupa una posición. Las posiciones se numeran del cero en adelante. Empezando con las unidades que ocupan la posición 0, las decenas la posición 1, las centenas la posición 2, y finalmente, los millares la posición 3. Otra manera de escribir este número es desglosando cada uno de ellos usando potencias de diez de la siguiente manera :

$$2943 = 2 \times 10^3 + 9 \times 10^2 + 4 \times 10^1 + 3 \times 10^0$$

Es decir, según la posición del dígito decimal, este se multiplica por una potencia de