

CONCETTI DI INFORMATICA E FONDAMENTI DI PYTHON

Prezzo di listino
Prezzo 54,00 €
: **Prezzo a te riservato**
51,30 €



Reparto	Libri
Autore	AA.VV.
Tipologia	Libri
Editore	Maggioli
Data pubblicazione	27 lug 2019
Pagine	762
Edizione	2

Descrizione

Questo volume è dedicato alla programmazione con l'utilizzo di Python, un linguaggio molto diffuso tra i professionisti grazie alla sua potenza e semplicità sintattica, e di uso sempre più frequente anche in ambito universitario. Il testo accompagna il lettore all'acquisizione degli strumenti concettuali classici della programmazione strutturata e introduce alla programmazione ad oggetti, caratteristica del linguaggio Python, presentando gli argomenti - oggetti, classi, ereditarietà, incapsulamento, polimorfismo - con chiarezza e completezza. La nuova edizione, interamente aggiornata e rivista, presenta alcune interessanti librerie che guidano alla soluzione di problemi reali, come la realizzazione di grafici, l'elaborazione di file audio e di fogli elettronici in formato CSV, l'invio di messaggi di posta elettronica.

Il volume è arricchito da casi svolti che permettono di elaborare strategie di problem solving, esercizi di approfondimento teorico graduati per difficoltà e numerosi problemi di programmazione. Il libro è un ideale riferimento per un corso introduttivo di programmazione basato su Python e, per la sua particolare comprensibilità ed efficacia didattica, è anche un ottimo strumento di apprendimento in autonomia.

Capitolo 1 - Introduzione

Obiettivi del capitolo

- 1.1 Computer e programmi
- 1.2 L'anatomia di un computer
- 1.3 Il linguaggio di programmazione Python
- 1.4 L'ambiente di programmazione
- 1.5 Analisi del primo programma
- 1.6 Errori
- 1.7 Soluzione di problemi: progettazione di algoritmi

Riepilogo del capitolo

Esercizi di ripasso

Esercizi di programmazione

Capitolo 2 - Programmare con numeri e stringhe

Obiettivi del capitolo

- 2.1 Variabili
 - 2.1.1 Definire variabili
 - 2.1.2 Tipi di numeri
 - 2.1.3 Nomi delle variabili
 - 2.1.4 Costanti
 - 2.1.5 Commenti
- 2.2 Aritmetica
 - 2.2.1 Operatori aritmetici elementari

- 2.2.2 Potenze
- 2.2.3 Divisione intera e resto
- 2.2.4 Invocare funzioni
- 2.2.5 Funzioni matematiche
- 2.3 Soluzione di problemi: prima lo si fa a mano
- 2.4 Stringhe
 - 2.4.1 Il tipo di dato stringa
 - 2.4.2 Concatenazione e ripetizione
 - 2.4.3 Conversioni tra numeri e stringhe
 - 2.4.4 Stringhe e caratteri
 - 2.4.5 Metodi per stringhe
- 2.5 Dati in ingresso e in uscita
 - 2.5.1 Dati forniti in ingresso dall'utente
 - 2.5.2 Dati in ingresso di tipo numerico
 - 2.5.3 Impaginazione dei risultati
- 2.6 Grafica: semplici disegni
 - 2.6.1 Creare una finestra
 - 2.6.2 Linee e poligoni
 - 2.6.3 Colori e riempimento di forme
 - 2.6.4 Ovali, cerchi e testi
- Riepilogo del capitolo
- Esercizi di ripasso
- Esercizi di programmazione
- Capitolo 3 - Decisioni
- Obiettivi del capitolo
- 3.1 L'enunciato if
- 3.2 Operatori relazionali
- 3.3 Diramazioni annidate
- 3.4 Alternative multiple
- 3.5 Soluzione di problemi: diagrammi di flusso
- 3.6 Soluzione di problemi: casi di prova
- 3.7 Variabili booleane e operatori
- 3.8 Analisi di stringhe
- 3.9 Applicazione: validità dei dati in ingresso
- Riepilogo del capitolo
- Esercizi di ripasso

Esercizi di programmazione

Capitolo 4 - Cicli

Obiettivi del capitolo

4.1 Il ciclo while

4.2 Soluzione di problemi: tenere traccia dell'esecuzione

4.3 Applicazione: elaborazione di valori sentinella

4.4 Soluzione di problemi: storyboard

4.5 Algoritmi che usano cicli

4.5.1 Calcolo di somma e valor medio

4.5.2 Conteggio di valori che soddisfano una condizione

4.5.3 Richiesta ripetuta fino al raggiungimento di un obiettivo

4.5.4 Valore massimo e minimo

4.5.5 Confronto di valori consecutivi

4.6 Il ciclo for

4.7 Cicli annidati

4.8 Elaborare stringhe

4.8.1 Contare le corrispondenze

4.8.2 Trovare tutte le corrispondenze

4.8.3 Trovare la prima o l'ultima corrispondenza

4.8.4 Determinare se una stringa è valida

4.8.5 Costruire una nuova stringa

4.9 Applicazione: numeri casuali e simulazioni

4.9.1 Generare numeri casuali

4.9.2 Simulare lanci di un dado

4.9.3 Il metodo Monte Carlo

4.10 Grafica: elaborazione di immagini digitali

4.10.1 Applicazione di filtri

4.10.2 Apportare modifiche a un'immagine

4.11 Soluzione di problemi: iniziare con un caso semplice

Riepilogo del capitolo

Esercizi di ripasso

Esercizi di programmazione

Capitolo 5 - Funzioni

Obiettivi del capitolo

5.1 Funzioni come scatole nere

5.2 Realizzazione e collaudo di funzioni

- 5.2.1 Realizzazione di una funzione
- 5.2.2 Collaudo di una funzione
- 5.2.3 Programmi che contengono funzioni
- 5.3 Passaggio di parametri
- 5.4 Valori restituiti
- 5.5 Funzioni che non restituiscono un valore
- 5.6 Soluzione di problemi: funzioni riutilizzabili
- 5.7 Soluzione di problemi: miglioramenti successivi
- 5.8 Ambito di visibilità delle variabili
- 5.9 Grafica: progettazione di strumenti per l'elaborazione d'immagini
 - 5.9.1 Per iniziare
 - 5.9.2 Confrontare immagini
 - 5.9.3 Modificare la luminosità di un'immagine
 - 5.9.4 Ruotare un'immagine
 - 5.9.5 Usare il toolkit
- 5.10 Funzioni ricorsive (argomento facoltativo)

Riepilogo del capitolo

Esercizi di ripasso

Esercizi di programmazione

Capitolo 6 - Liste

Obiettivi del capitolo

- 6.1 Proprietà basilari delle liste
 - 6.1.1 Creazione di una lista
 - 6.1.2 Accesso agli elementi di una lista
 - 6.1.3 Scansione di una lista
 - 6.1.4 Riferimenti a una lista
- 6.2 Operazioni con le liste
 - 6.2.1 Aggiungere elementi
 - 6.2.2 Inserire un elemento
 - 6.2.3 Cercare un elemento
 - 6.2.4 Eliminare un elemento
 - 6.2.5 Concatenazione e replicazione
 - 6.2.6 Verifica di uguaglianza
 - 6.2.7 Somma, massimo, minimo e ordinamento
 - 6.2.8 Copiatura
- 6.3 Algoritmi elementari che operano su liste

- 6.3.1 Riempimento
- 6.3.2 Elaborazione di tutti gli elementi
- 6.3.3 Separatori tra elementi
- 6.3.4 Massimo e minimo
- 6.3.5 Ricerca lineare
- 6.3.6 Raccolta e conteggio di corrispondenze
- 6.3.7 Eliminazione di corrispondenze
- 6.3.8 Scambio di elementi
- 6.3.9 Acquisizione di dati
- 6.4 Utilizzo di liste nelle funzioni
- 6.5 Soluzione di problemi: adattamento di algoritmi
- 6.6 Soluzione di problemi: scoprire algoritmi facendo esperimenti concreti
- 6.7 Tabelle
 - 6.7.1 Creazione di tabelle
 - 6.7.2 Accesso agli elementi
 - 6.7.3 Individuazione degli elementi adiacenti
 - 6.7.4 Sommare righe e colonne
 - 6.7.5 Funzioni che usano tabelle
- Riepilogo del capitolo
- Esercizi di ripasso
- Esercizi di programmazione
- Capitolo 7 - Eccezioni e file
- Obiettivi del capitolo
- 7.1 Leggere e scrivere file di testo
 - 7.1.1 Aprire un file
 - 7.1.2 Leggere un file
 - 7.1.3 Scrivere un file
 - 7.1.4 Un esempio di elaborazione di file
- 7.2 Testo in ingresso e in uscita
 - 7.2.1 Scandire le righe di un file
 - 7.2.2 Leggere parole
 - 7.2.3 Leggere caratteri
 - 7.2.4 Leggere dati composti (record)
- 7.3 Argomenti sulla riga di comando
- 7.4 File binari e accesso casuale (argomento facoltativo)
 - 7.4.1 Leggere e scrivere file binari

- 7.4.2 Accesso casuale
- 7.4.3 File contenenti immagini
- 7.4.4 Elaborare file BMP
- 7.5 Gestione delle eccezioni
 - 7.5.1 Sollevare eccezioni
 - 7.5.2 Gestire eccezioni
 - 7.5.3 La clausola finally
- 7.6 Applicazione: gestione degli errori di acquisizione

Riepilogo del capitolo

Esercizi di ripasso

Esercizi di programmazione

Capitolo 8 - Insiemi e dizionari

Obiettivi del capitolo

8.1 Insiemi

8.1.1 Creare e usare insiemi

8.1.2 Aggiungere e rimuovere elementi

8.1.3 Sottoinsiemi

8.1.4 Unione, intersezione e differenza tra insiemi

8.2 Dizionari

8.2.1 Creare dizionari

8.2.2 Accedere ai valori in un dizionario

8.2.3 Aggiungere e modificare coppie

8.2.4 Eliminare coppie

8.2.5 Scandire gli elementi di un dizionario

8.3 Strutture complesse

8.3.1 Un dizionario di insiemi

8.3.2 Un dizionario di liste

Riepilogo del capitolo

Esercizi di ripasso

Esercizi di programmazione

Capitolo 9 - Oggetti e classi

Obiettivi del capitolo

9.1 Programmazione orientata agli oggetti

9.2 Realizzare una classe semplice

9.3 Definire l'interfaccia pubblica di una classe

9.4 Progettare la rappresentazione dei dati

- 9.5 Costruttori
- 9.6 Realizzare metodi
- 9.7 Collaudare una classe
- 9.8 Soluzione di problemi: tenere traccia di oggetti
- 9.9 Soluzione di problemi: schemi ricorrenti per oggetti
 - 9.9.1 Tenere aggiornata una quantità totale
 - 9.9.2 Contare eventi
 - 9.9.3 Collezionare valori
 - 9.9.4 Gestire caratteristiche di oggetti
 - 9.9.5 Definire oggetti con più stati distinti
 - 9.9.6 Descrivere la posizione di un oggetto
- 9.10 Riferimenti a oggetti
 - 9.10.1 Riferimenti condivisi
 - 9.10.2 Il riferimento None
 - 9.10.3 Il riferimento self
 - 9.10.4 La vita di un oggetto
- 9.11 Applicazione: progettare la classe Fraction
 - 9.11.1 Il progetto della classe Fraction
 - 9.11.2 Il costruttore
 - 9.11.3 Metodi speciali
 - 9.11.4 Operazioni aritmetiche
 - 9.11.5 Operazioni logiche
- Riepilogo del capitolo
- Esercizi di ripasso
- Esercizi di programmazione
- Capitolo 10 - Ereditarietà
- Obiettivi del capitolo
- 10.1 Gerarchie di ereditarietà
- 10.2 Realizzare sottoclassi
- 10.3 Invocare il costruttore della superclasse
- 10.4 Sovrascrivere metodi
- 10.5 Polimorfismo
- 10.6 Applicazione: una gerarchia di classi per forme geometriche
 - 10.6.1 La classe di base
 - 10.6.2 Forme elementari
 - 10.6.3 Gruppi di forme

Riepilogo del capitolo

Esercizi di ripasso

Esercizi di programmazione

Capitolo 11 - Ricorsione

Obiettivi del capitolo

11.1 Di nuovo i numeri triangolari

11.2 Soluzione di problemi: pensare ricorsivamente

11.3 Funzioni ausiliarie ricorsive

11.4 L'efficienza della ricorsione

11.5 Permutazioni

11.6 Backtracking

11.7 Ricorsione mutua

Riepilogo del capitolo

Esercizi di ripasso

Esercizi di programmazione

Capitolo 12 - Ordinamento e ricerca

Obiettivi del capitolo

12.1 Ordinamento per selezione

12.2 Misurazione delle prestazioni dell'ordinamento per selezione

12.3 Analisi delle prestazioni dell'algoritmo di ordinamento per selezione

12.4 Ordinamento per fusione (merge sort)

12.5 Analisi dell'algoritmo di ordinamento per fusione

12.6 Effettuare ricerche

12.6.1 Ricerca lineare

12.6.2 Ricerca binaria

12.7 Soluzione di problemi: stimare il tempo di esecuzione di un algoritmo

12.7.1 Algoritmi lineari

12.7.2 Algoritmi quadratici

12.7.3 Lo schema triangolare

12.7.4 Algoritmi logaritmici

Riepilogo del capitolo

Esercizi di ripasso

Esercizi di programmazione

Indice analitico

Appendice A - Linguaggio Python: operatori

Appendice B - Linguaggio Python: parole riservate

Appendice C - Linguaggio Python: libreria standard

Appendice D - I sottoinsiemi Basic Latin e Latin-1 di Unicode

Appendice E - Numeri binari e operazioni tra bit

Appendice F - Compendio di HTML

Appendice G - Compendio di Python

Glossario

Rimaniamo a disposizione per qualsiasi ulteriore chiarimento allo 0461.232337 o 0461.980546

oppure via mail a : servizioclienti@libriprofessionali.it

www.LibriProfessionali.it è un sito di Scala snc Via Solteri, 74 38121 Trento (Tn) P.Iva 01534230220

