

Prefazione all'edizione italiana

La settima edizione di questo testo è uscita negli Stati Uniti in contemporanea con la definizione dell'ultimo standard del linguaggio C, il cosiddetto C11. La nuova edizione recepisce questo nuovo standard, dedicandogli note specifiche e un'appendice on-line (Appendice F). Il nuovo testo rimane così un'opera completa e aggiornata, essendo insieme un libro sul linguaggio C e anche un libro di introduzione alla programmazione in generale. Il taglio è quello ormai collaudato dei libri a marchio Deitel sulla programmazione nei vari linguaggi, C, C++, Java, e così via. L'enfasi è posta sulle metodologie di programmazione adatte allo specifico linguaggio, che vengono presentate con un taglio orientato all'effettivo sviluppo di applicazioni, mediante codice funzionante e analisi dettagliata.

Dopo un'ampia trattazione introduttiva sui sistemi informatici e sullo stato dell'arte delle tecnologie informatiche, l'approccio metodologico si snoda partendo dalle tecniche di programmazione strutturata e procedendo con i costrutti del linguaggio che la supportano, fino a mostrare, con numerosi e corposi esempi ed esercizi, come costruire effettivamente sistemi software anche complessi. Nel corso della trattazione vengono affrontate ampiamente e in modo operativo le tematiche relative agli algoritmi e alle strutture di dati, fondamentali per un approccio professionale alla programmazione.

L'edizione italiana riguarda la prima parte del testo originale, che nella sua versione in lingua inglese comprende anche una seconda parte dedicata al linguaggio C++. L'opera si rivolge sia a coloro che operano in ambito tecnico e scientifico, per i quali può essere un valido riferimento, sia a coloro che vogliano intraprendere un percorso serio e rigoroso che li porti a saper programmare. Il testo è, pertanto, particolarmente adatto a essere utilizzato in corsi universitari riguardanti la programmazione, sia introduttivi che avanzati. A tal proposito, la prefazione degli autori indica diversi possibili percorsi di utilizzo del testo.

Agli studenti il C può apparire inizialmente un linguaggio "ostico", per via del suo ridotto livello di astrazione, ma una volta appresi, con un po' di dedizione, i concetti fondamentali, insieme a una buona pratica di programmazione, si viene ripagati dalla possibilità di scrivere programmi compatti, efficienti ed estremamente portabili. Riguardo a ciò, nonostante diverse standardizzazioni, il C è un linguaggio sostanzialmente stabile da circa quarant'anni e per questo utilizzabile senza grossi stravolgimenti in qualsiasi classe di elaboratori.

In merito poi alla collocazione della programmazione in linguaggio C nell'ambito degli insegnamenti universitari dei corsi di laurea in informatica e in ingegneria informatica, possiamo dire che tale problematica va inquadrata nell'ampio dibattito degli scorsi anni relativo al primo linguaggio con cui introdurre la programmazione. Subito dopo l'esplosione del web come fenomeno sociale, alcune scuole di pensiero hanno ritenuto opportuno iniziare con un approccio più di tipo sistemistico e basato sull'astrazione, attraverso l'utilizzo di linguaggi come Java che introducono subito la metodologia ad oggetti e sono orientati al web. Oggi si assiste sempre di più a un ripensamento verso un percorso più tradizionale, che parte dalla conoscenza dell'architettura hardware e procede a tappe verso modelli più astratti e sistemistici. Il linguaggio C è il linguaggio ad alto livello giusto da cui partire, poiché da un lato è direttamente comprensibile in riferimento alla macchina su cui girano i programmi, e dall'altro permette di introdurre facilmente i costrutti di più alto livello, che si possono successivamente, in modo più appropriato, sviluppare nel contesto di

altri linguaggi. La conoscenza della programmazione in C è anche propedeutica per i corsi successivi che riguardano il software di sistema, proprio perché è il linguaggio in genere più usato a tale scopo ed è quello sicuramente più adatto per gli aspetti legati alle prestazioni.

Personalmente ho accolto con piacere l'invito di Pearson Italia di curare la traduzione di questa nuova edizione, proprio perché si tratta di un testo di grande valore sia dal punto di vista tecnico che didattico, la cui impostazione è maturata attraverso sei edizioni precedenti. In secondo luogo, perché tratta argomenti preziosi e fondamentali per ogni informatico, quali il linguaggio C e la programmazione strutturata. Essi fanno parte integrante di quella rivoluzione informatica degli anni Settanta, vissuta professionalmente da molti di noi; in quegli anni la tecnologia rendeva il computer oggetto pervasivo e accessibile a tutti, e dai sistemi chiusi e proprietari si passava ai sistemi aperti. Nessuno si era allora reso conto di quelli che sarebbero stati i risvolti sociali e umani di quella rivoluzione, seguita poi da quella di Internet e del web negli anni Novanta.

Concludo col ringraziare la traduttrice Ivana La Rosa per il paziente e meticoloso lavoro, lo staff di Pearson Italia, in particolare Valentina Favata e Alessandra Piccardo, per la grande disponibilità nel condividere le scelte di tipo editoriale e per la puntuale e competente assistenza nella verifica delle bozze, nonché i colleghi Daniele Peri e Filippo Vella per aver letto il testo tradotto e per aver contribuito con i loro consigli a migliorarlo. Un grazie anche ai colleghi Giuseppe Lo Re e Valeria Seidita per i loro commenti e infine a Vincenzo Catanese per il suo entusiasmo nel convincermi a curare l'edizione italiana.

Salvatore Gaglio
Dipartimento di Ingegneria Chimica, Gestionale,
Informatica e Meccanica
Università degli Studi di Palermo

Prefazione

Benvenuti nel linguaggio di programmazione C. Questo libro presenta tecnologie informatiche fondamentali e d'avanguardia per studenti universitari, ricercatori e professionisti che operano nello sviluppo del software.

Al centro del libro sta l'“approccio live-code” (letteralmente “codice dal vivo”) proprio del marchio Deitel. I concetti vengono presentati nel contesto di programmi completi e funzionanti, invece che con frammenti di codice. Ogni esempio di codice è seguito da uno o più esempi di esecuzione. Tutto il codice sorgente è disponibile sui siti www.deitel.com/books/chtp7/ e www.pearsonhighered.com/deitel. Usate questo codice sorgente *per far eseguire ogni programma* mentre lo studiate.

Riteniamo che questo libro e il suo materiale di supporto vi offriranno un'introduzione al linguaggio C istruttiva, stimolante e piacevole. Se durante la lettura avete domande da porre, mandate una e-mail all'indirizzo deitel@deitel.com. Risponderemo prontamente. Per aggiornamenti sul libro visitate il sito www.deitel.com/books/chtp7/, unitevi alle nostre comunità su Facebook (www.deitel.com/deitelfan), Twitter (@deitel) e Google+ ([gplus.to/deitel](https://plus.google.com/+deitel)) e iscrivetevi alla newsletter *Deitel® Buzz Online* (www.deitel.com/newsletter/subscribe.html).

Caratteristiche nuove e aggiornate

- **Trattazione del Nuovo C standard.** La precedente edizione del libro era conforme al “C standard” e comprendeva un'appendice particolareggiata sullo standard C99. Il Nuovo C standard è stato approvato poco prima che il libro fosse pubblicato. Il nuovo standard include sia il C99 che il più recente C1X, ora detto C11 o semplicemente “il C standard” per via della sua approvazione nel 2011. Il supporto per il nuovo standard varia a seconda del compilatore. La grande maggioranza dei nostri lettori usa o il compilatore GNU gcc – che supporta diverse delle caratteristiche chiave nel nuovo standard – o il compilatore Visual C++ di Microsoft. Microsoft supporta solo un sottoinsieme limitato delle caratteristiche che sono state aggiunte al C nel C99 e nel C11, principalmente quelle richieste anche dal C++ standard. Per venire incontro a tutti i nostri lettori, abbiamo incluso la descrizione del nuovo standard in paragrafi opzionali, facili da usare o da tralasciare, e nell'Appendice F. Per aderire al nuovo C standard abbiamo anche sostituito varie funzionalità disapprovate con versioni più nuove e generalmente preferite.
- **Nuovo Capitolo 1.** Il nuovo Capitolo 1 intende attrarre gli studenti con dati statistici interessanti allo scopo di appassionarli allo studio e alla programmazione dei computer. Il capitolo include una tabella relativa ad alcune delle ricerche rese possibili dai computer e da Internet, una descrizione delle attuali tendenze della tecnologia e delle architetture hardware, un'illustrazione della gerarchia dei dati, un nuovo paragrafo sui social network, una tabella sulle pubblicazioni commerciali e tecnologiche e sui siti web che vi aiuteranno a rimanere aggiornati con le ultime notizie e tendenze in ambito tecnologico e, infine, esercizi aggiornati. Abbiamo incluso paragrafi che illustrano come far eseguire un programma in C dalla linea di comando su Windows di Microsoft, Linux e Mac OS X.

- **Paragrafi sulla programmazione sicura in C.** In molti capitoli riguardanti la programmazione in C abbiamo aggiunto alcune note sulla programmazione sicura in C. Per maggiori dettagli consultate il paragrafo di questa prefazione “Una nota sulla programmazione sicura in C”.
- **Attenzione agli aspetti relativi alle prestazioni.** Il C e il C++ sono i linguaggi preferiti dai progettisti di applicazioni caratterizzate da prestazioni intensive come sistemi operativi, sistemi in tempo reale, sistemi embedded (incorporati) e sistemi di comunicazione. Per questo motivo ci concentriamo intensamente sugli aspetti relativi alle prestazioni.
- **Esercizi per “fare la differenza”.** Vi incoraggiamo a usare i computer e Internet per fare ricerche e risolvere problemi di reale interesse. Questi esercizi intendono aumentare la consapevolezza sulle questioni importanti che il mondo sta affrontando, e la nostra speranza è che li affrontiate in base ai vostri valori, alle vostre idee politiche e ai vostri principi.
- **Tutto il codice testato su Windows e Linux.** Abbiamo testato ogni esempio e ogni programma degli esercizi usando Visual C++ e GNU gcc, rispettivamente in Windows e in Linux.
- **Ordinamento: uno sguardo più approfondito.** Le tecniche di ordinamento mettono in ordine i dati sulla base di una o più chiavi di ordinamento. Iniziamo la nostra presentazione delle tecniche di ordinamento con un semplice algoritmo nel Capitolo 6. Nell’Appendice E offriamo una trattazione più approfondita. Prendiamo in considerazione diversi algoritmi e li confrontiamo, tenendo conto del loro consumo di memoria e delle loro esigenze di calcolo. A questo scopo introduciamo la notazione “O grande”, che indica lo sforzo che può essere richiesto a un algoritmo per risolvere un problema. Mediante esempi ed esercizi, l’Appendice E analizza l’ordinamento per selezione, l’ordinamento per inserzione, l’ordinamento per fusione ricorsivo, l’ordinamento per selezione ricorsivo, il bucket sort e il quicksort ricorsivo. L’ordinamento è un problema interessante, perché le varie tecniche di ordinamento conseguono lo stesso risultato finale ma possono variare enormemente per quanto riguarda il loro consumo di memoria, di tempo di CPU e di altre risorse di sistema.
- **Esercizi di programmazione con titolo.** Tutti gli esercizi di programmazione hanno un titolo per consentire ai docenti di scegliere opportunamente i compiti per le loro classi.
- **Ordine di valutazione.** Abbiamo aggiunto alcune note riguardo agli aspetti relativi all’ordine di valutazione.
- **Esercizi aggiuntivi.** Abbiamo aggiunto altri esercizi sui puntatori a funzioni. Abbiamo aggiunto anche un esercizio per apportare miglioramenti all’esempio sulla ricorsione di Fibonacci (uso della ricorsione di coda).
- **Libreria Standard del C.** Il Paragrafo 1.7 menziona il sito Dinkumware di P.J. Plauger (www.dinkumware.com/manuals/default.aspx) dove gli studenti possono trovare una documentazione completa e facilmente consultabile sulle funzioni della Libreria Standard del C.

Una nota sulla programmazione sicura in C

In tutto questo libro ci concentriamo sui *fondamenti* della programmazione in C. Quando scriviamo ogni libro della serie “How to Program”, cerchiamo nei documenti relativi agli standard del linguaggio gli aspetti che secondo noi i principianti devono imparare in un corso iniziale di programmazione e gli aspetti che coloro che hanno esperienza di programmazione devono sapere per *cominciare* a lavorare in quel linguaggio. Dobbiamo trattare anche i fondamenti della programmazione e i fondamenti dell’informatica per programmatori principianti, i nostri principali lettori.

Le tecniche di codifica *a livello industriale* di un qualsiasi linguaggio di programmazione vanno oltre gli scopi di un libro di testo introduttivo. Per questa ragione, i nostri paragrafi sulla

programmazione sicura in C presentano alcuni problemi e tecniche chiave e forniscono collegamenti e rimandi, così che possiate proseguire nell'apprendimento.

L'esperienza ha dimostrato che è difficile costruire sistemi di portata industriale che resistano ad attacchi di virus, worm, ecc. Oggi, via Internet, attacchi del genere possono essere istantanei e globali. Le vulnerabilità del software derivano spesso da semplici problemi di programmazione. Includere aspetti relativi alla sicurezza nel software dall'inizio del ciclo di sviluppo può ridurre sensibilmente i costi e le vulnerabilità.

Il centro di coordinamento CERT[®] (www.cert.org) è stato creato per analizzare e rispondere prontamente agli attacchi. Il CERT (Computer Emergency Response Team) pubblica e promuove standard di codifica sicura per aiutare i programmatori in C (e non solo) a implementare sistemi di portata industriale che evitino pratiche di programmazione che espongono i sistemi agli attacchi. Gli standard del CERT si evolvono quando sorgono nuovi problemi di sicurezza.

Abbiamo aggiornato il nostro codice (com'è giusto per un libro introduttivo) per conformarlo alle varie raccomandazioni del CERT. Qualora doveste realizzare sistemi in C in ambito industriale, prendete in considerazione la lettura di testi informativi aggiornati. Le linee guida del CERT sono disponibili online al sito www.securecoding.cert.org. Robert Seacord, un revisore tecnico di questo libro, ha fornito raccomandazioni specifiche per ognuno dei nostri nuovi paragrafi sulla programmazione sicura in C. Robert Seacord è Secure Coding Manager al CERT presso il Software Engineering Institute (SEI) della Carnegie Mellon University e anche professore aggiunto alla School of Computer Science della Carnegie Mellon University.

I paragrafi sulla programmazione sicura in C alla fine dei Capitoli dal 2 al 13 prendono in esame molti argomenti importanti, compresi il controllo degli overflow aritmetici, l'uso di tipi interi senza segno, le nuove funzioni più sicure nell'Annex K del C standard, l'importanza del controllo delle informazioni sullo stato restituite dalle funzioni della libreria standard, il controllo degli intervalli, la generazione sicura di numeri casuali, il controllo dei confini di un array, le tecniche per la prevenzione degli overflow dei buffer, la validazione dell'input, le tecniche per evitare comportamenti indefiniti, la scelta di funzioni che restituiscono informazioni sullo stato invece delle funzioni simili che non lo fanno, la garanzia che i puntatori siano sempre NULL o contengano indirizzi validi, l'uso delle funzioni del C rispetto all'uso delle macro del preprocessore, e altro ancora.

Materiale sul web

Questo libro è supportato da importante materiale online. Il Companion Website del libro (www.pearsonhighered.com/deitel) contiene il codice sorgente per tutti gli esempi di codice e le seguenti appendici in formato PDF con testo accessibile:

- Appendix F, Additional Features of the C99 and C11 Standards
- Appendix G, Using the Visual Studio Debugger
- Appendix H, Using the GNU Debugger

Diagramma di dipendenza

La Figura 1 mostra le dipendenze tra i capitoli del libro per permettere agli istruttori di pianificare i programmi dei loro corsi. Il testo è adatto per i corsi di programmazione in C di livello base e intermedio.

Diagramma di dipendenza dei capitoli del libro

[Nota: le frecce che puntano verso un capitolo indicano le dipendenze di quel capitolo.]

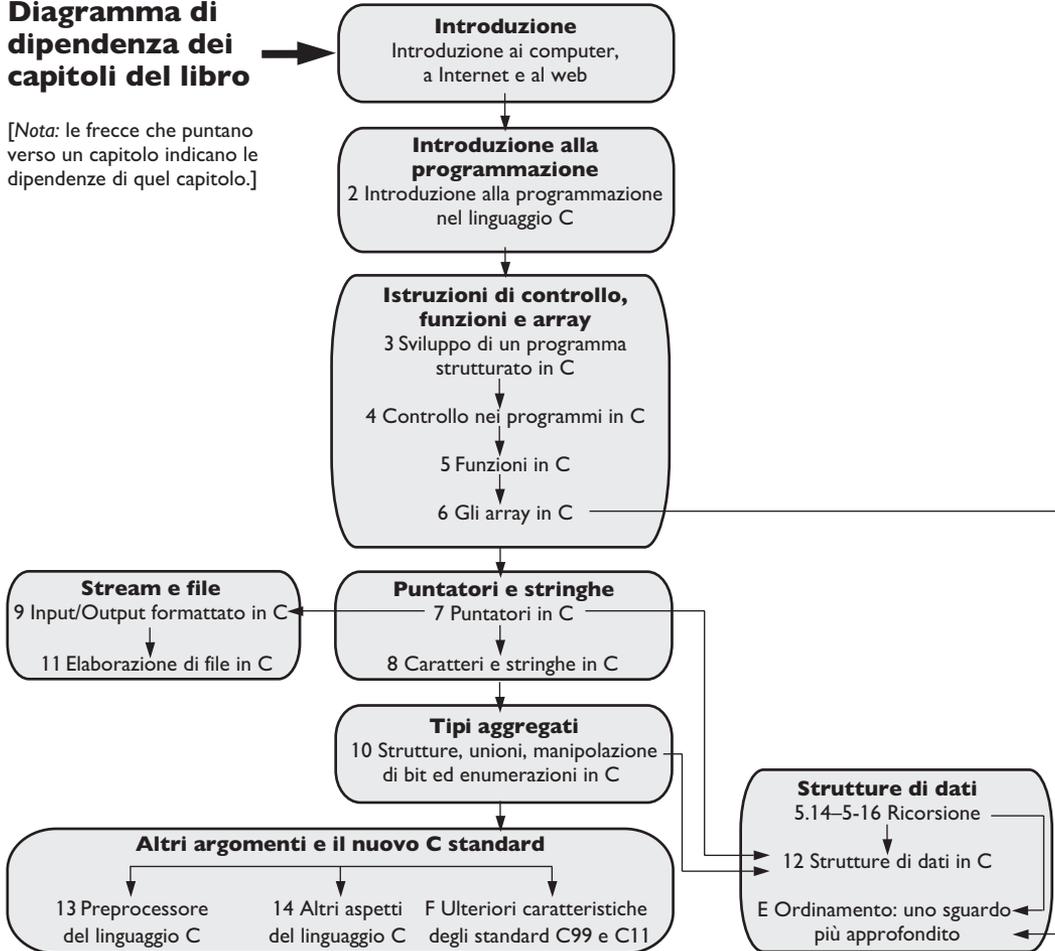


Figura 1 Diagramma di dipendenza dei capitoli del libro.

Approccio didattico

Il libro contiene una ricca raccolta di esempi. Poniamo l’attenzione su una buona ingegneria del software e sull’importanza della chiarezza dei programmi.

Evidenziazione del testo in base alla sintassi. Per migliorare la leggibilità, evidenziamo il codice in base alla sintassi, analogamente a come la maggior parte degli IDE (*Interactive Development Environment*, “ambienti di sviluppo interattivi”) e degli editor di codice colorano il testo. Le nostre convenzioni di evidenziazione sono queste:

```

i commenti appaiono in questo modo
le parole chiave appaiono in questo modo
le costanti e i valori letterali appaiono in questo modo
tutto il resto del codice appare in questo modo
  
```

Dare rilievo al codice. Racchiudiamo il codice chiave entro rettangoli grigi.

Uso di caratteri diversi per mettere in risalto. Per un facile riferimento, mettiamo in grassetto i termini chiave e, nell'indice, il numero della pagina dove i termini vengono definiti. Mettiamo in risalto i componenti grafici che compaiono sullo schermo con il carattere **Helvetica grassetto** (es. il menu **File**) e il testo dei programmi in C con il carattere **Consolas** (es. `int x = 5;`).

Obiettivi. All'inizio di ogni capitolo c'è un elenco degli obiettivi che ci prefiggiamo di raggiungere.

Illustrazioni/Figure. Comprendono vari diagrammi, tabelle, disegni, programmi e output di programmi.

Consigli sulla programmazione. Includiamo vari consigli sulla programmazione per aiutarvi a concentrarvi su aspetti importanti dello sviluppo di un programma. Questi consigli e queste pratiche rappresentano il meglio che abbiamo raccolto da un'esperienza combinata di programmazione e di insegnamento che dura da settant'anni.



Buone pratiche di programmazione

Le Buone pratiche di programmazione richiamano l'attenzione sulle tecniche che vi aiuteranno a produrre programmi più chiari, più comprensibili e più mantenibili.



Errori comuni di programmazione

Dare rilievo agli Errori comuni di programmazione riduce la probabilità che li commettiate.



Prevenzione di errori

Questi consigli contengono suggerimenti per scoprire gli errori e rimuoverli dai vostri programmi; molti di essi descrivono aspetti del C che permettono di evitare determinati errori nei programmi già dalla prima stesura.



Prestazioni

Questi consigli mettono in evidenza le tecniche per far sì che i vostri programmi siano eseguiti più velocemente o per minimizzare la quantità di memoria occupata.



Portabilità

I consigli sulla portabilità vi aiutano a scrivere codice che potrà essere eseguito su una varietà di piattaforme.



Osservazioni di ingegneria del software

Le Osservazioni di ingegneria del software danno risalto agli aspetti architetturali e progettuali inerenti alla costruzione di sistemi software, specialmente di sistemi a larga scala.

Riepilogo di ogni capitolo. Includiamo in ogni capitolo un riepilogo paragrafo per paragrafo sotto forma di elenco puntato.

Esercizi di autovalutazione e risposte. Sono inclusi molti esercizi di autovalutazione con le relative risposte come ausilio allo studio personale.

Esercizi. Ogni capitolo si conclude con un insieme corposo di esercizi che riguardano:

- semplici richiami a terminologia e concetti importanti
- l'identificazione di errori negli esempi di codice
- la scrittura di singole istruzioni di programmi
- la scrittura di funzioni in C
- la scrittura di programmi completi
- la realizzazione di progetti

Indice. Abbiamo incluso un ampio indice, utile specialmente quando si usa il libro come riferimento.

Software utilizzato

Abbiamo scritto e provato i programmi che compaiono nel testo usando la versione gratuita di Microsoft Visual C++ Express Edition (che può compilare programmi sia in C che in C++ e può essere scaricata dal sito www.microsoft.com/express/downloads/) e il compilatore GNU C (gcc.gnu.org/install/binaries.html) che è già installato su gran parte dei sistemi Linux e può essere installato su Mac OS X e sui sistemi Windows. Apple include GNU C e C++ nei propri strumenti di sviluppo Xcode che gli utenti del Mac OS X possono scaricare dal sito developer.apple.com/technologies/tools/xcode.html.

Per altri compilatori gratuiti per il linguaggio C visitate i siti:

```
www.thefreecountry.com/compilers/cpp.shtml
www.compilers.net/Dir/Compilers/CCpp.htm
www.freebyte.com/programming/cpp/#cppcompilers
en.wikipedia.org/wiki/List_of_compilers#C.2B.2B_compilers
```

Risorse per istruttori

I seguenti supplementi sono disponibili *solo per istruttori qualificati* tramite l'Instructor Resource Center di Pearson Education (www.pearsonhighered.com):

- **PowerPoint® slide** contenenti tutti i codici e le figure del testo, più altri elementi che sintetizzano i punti chiave.
- **Test Item File** contenente domande a scelta multipla (all'incirca due per paragrafo).
- **Manuale delle soluzioni** con soluzioni per la maggior parte degli esercizi di fine capitolo. Controllate il centro risorse per i docenti per determinare quali esercizi hanno soluzioni.

Per favore, non scrivete per richiedere l'accesso al Pearson Instructor's Resource Center. L'accesso è riservato ai docenti universitari che adottano il libro nel loro corso. I docenti possono ottenere l'accesso solo tramite i rappresentanti Pearson Italia. Se non siete registrati come docenti, contattate il vostro consulente accademico Pearson o visitate il sito <http://hpe.pearson.it/docenti.php> e cliccate su "Contatti i Consulenti Accademici".

Non vengono fornite soluzioni per gli esercizi riguardanti "progetti". Consultate il nostro Programming Projects Resource Center per numerosi altri esercizi e progetti (www.deitel.com).

Ringraziamenti

Vorremmo ringraziare Abbey Deitel e Barbara Deitel per le lunghe ore dedicate a questo progetto. Siamo fortunati ad aver lavorato con la squadra di professionisti dell'editoria della Pearson. Apprezziamo la consulenza, l'esperienza e l'energia di Michael Hirsch, redattore capo per la sezione di Computer Science. Carole Snyder e Bob Engelhardt hanno fatto un lavoro meraviglioso occupandosi, rispettivamente, della revisione e dei processi di produzione.

Revisori della settima edizione

Desideriamo esprimere gratitudine per gli sforzi dei nostri revisori. In tempi molto stretti hanno esaminato attentamente il testo e i programmi, fornendo innumerevoli suggerimenti per migliorarne la presentazione: Dr. John F. Doyle (Indiana University Southeast), Hemanth H.M. (Ingegnere del Software presso SonicWALL), Vytautas Leonavicius (Microsoft), Robert Seacord (Secure Coding Manager presso il SEI/CERT, autore di *The CERT C Secure Coding Standard* e José Antonio González Seco (Parlamento dell'Andalusia).

Revisori di altre recenti edizioni

William Albrecht (University of South Florida), Ian Barland (Radford University), Ed James Beckham (Altera), John Benito (Blue Pilot Consulting, Inc. and Convener of ISOWG14 — il Gruppo di Lavoro responsabile del C Programming Language Standard), Alireza Fazelpour (Palm Beach Community College), Mahesh Hariharan (Microsoft), Kevin Mark Jones (Hewlett Packard), Lawrence Jones, (UGS Corp.), Don Kostuch (In-dependent Consultant), Xiaolong Li (Indiana State University), William Mike Miller (Edison Design Group, Inc.), Tom Rethard (University of Texas di Arlington), Benjamin Seyfarth (University of Southern Mississippi), Gary Sibbitts (St. Louis Community College di Meramec), William Smith (Tulsa Community College) e Douglas Walls (Senior Staff Engineer, C compiler, Sun Microsystems).

Bene, avete capito! Il C è un formidabile linguaggio di programmazione che vi aiuterà a scrivere velocemente e bene programmi ad alte prestazioni. Il C è facilmente scalabile fino al settore dello sviluppo di sistemi d'impresa e consente alle aziende di costruire i propri sistemi informativi indispensabili per le strategie aziendali. Dopo che avrete letto il libro, apprezzeremo sinceramente i vostri commenti, le critiche, le correzioni e i suggerimenti per migliorare il testo. Si prega di indirizzare tutta la corrispondenza a:

deitel@deitel.com

Risponderemo prontamente. Inviare le correzioni e i chiarimenti al sito:

www.deitel.com/books/chtp7/

Speriamo che vi piaccia lavorare con questo libro così come a noi è piaciuto scriverlo!

Paul Deitel
Harvey Deitel
Gennaio 2012

Gli autori

Paul Deitel, CEO e Chief Technical Officer della Deitel & Associates, Inc., si è laureato al MIT, dove ha studiato Information Technology. Tramite la Deitel & Associates, Inc., ha tenuto centinaia di corsi di programmazione per clienti del ramo industriale, compresi Cisco, IBM, Siemens, Sun Microsystems, Dell, Lucent Technologies, Fidelity, NASA al Kennedy Space Center, il National Severe Storm Laboratory, White Sands Missile Range, Rogue Wave Software, Boeing, SunGard Higher Education, Stratus, Cambridge Technology Partners, One Wave, Hyperion Software, Adra Systems, Entergy, CableData Systems, Nortel Networks, Puma, iRobot, Invensys e molti altri. Paul Deitel e il suo coautore, Harvey M. Deitel, sono autori delle pubblicazioni (libri di testo, libri a carattere professionale e video) sui linguaggi di programmazione più venduti al mondo.

Harvey Deitel, Chairman e Chief Strategy Officer della Deitel & Associates, Inc., ha 50 anni di esperienza nel campo dei computer. Ha ottenuto le lauree di primo e secondo livello al MIT e un Ph.D. alla Boston University. Ha una vasta esperienza di insegnamento universitario ed è anche stato Direttore del Dipartimento di Informatica al Boston College prima di fondare nel 1991 la Deitel & Associates, Inc. con suo figlio Paul. Le pubblicazioni dei Deitel hanno ottenuto un riconoscimento internazionale, con traduzioni pubblicate in cinese, coreano, giapponese, tedesco, russo, spagnolo, francese, polacco, italiano, portoghese, greco, urdu e turco. Harvey Deitel ha tenuto centinaia di seminari sulla programmazione professionale per le più importanti aziende, istituzioni accademiche, organizzazioni governative e militari.

Formazione aziendale da parte di Deitel & Associates, Inc.

La Deitel & Associates, Inc., fondata da Paul Deitel e Harvey Deitel, è un'organizzazione riconosciuta a livello internazionale per la creazione di contenuti multimediali e interattivi, per la formazione aziendale e lo sviluppo di software, specializzata nei linguaggi di programmazione per computer, nella tecnologia a oggetti, nello sviluppo di app per Android e iPhone e nelle tecnologie per Internet e software per il web. L'azienda offre corsi di formazione, tenuti presso le sedi dei clienti in tutti il mondo, sui principali linguaggi di programmazione e sulle principali piattaforme, compresi C, C++, Visual C++[®], Java[™], Visual C#[®], Visual Basic[®], XML[®], Python[®], la tecnologia degli oggetti, la programmazione per Internet e per il web, lo sviluppo di app per Android, lo sviluppo di app in Objective-C per iPhone, oltre a un elenco crescente di ulteriori corsi di programmazione e di sviluppo del software. I clienti sono molte delle più grandi aziende del mondo, agenzie governative, settori dell'esercito e istituzioni accademiche.

Grazie ai suoi 36 anni di partnership editoriale con Prentice Hall/Pearson, Deitel & Associates, Inc., pubblica libri di testo universitari sulla programmazione avanzata, libri professionali e corsi video con *lezioni dal vivo*. Potete contattare Deitel & Associates, Inc., e gli autori al sito:

deitel@deitel.com

Per saperne di più su *Dive Into Series*[®] di Deitel, in particolare sull'offerta riguardante la formazione aziendale, visitate il sito:

www.deitel.com/training/

Per richiedere una proposta di formazione in loco con istruttori qualificati per la vostra azienda od organizzazione, inviate un'e-mail all'indirizzo deitel@deitel.com.

Chi desidera acquistare i libri di Deitel e i video delle *Lezioni dal vivo* può farlo tramite il sito www.deitel.com.