

Prefazione

Questo libro offre un'introduzione tradizionale alla scienza informatica utilizzando strumenti attuali. Impostata secondo un punto di vista scientifico, l'attività di programmazione viene presentata in modo accessibile, completo e approfondito. La programmazione è il tema centrale della scienza informatica ed è per questo che qui si insegnerà a programmare.

Sebbene questo testo abbia un'impostazione tradizionale, utilizza le tecniche più moderne in tre modi.

- *Il linguaggio di programmazione adottato è un subset di C++.* Sebbene C++ sia molto lontano dall'essere un linguaggio didattico, il suo utilizzo nell'insegnamento ha un significato pragmatico. C++ viene ampiamente utilizzato nell'industria del software. Ambienti di programmazione pratici ed economici sono disponibili per tutte le piattaforme principali. C++ è abbastanza esaustivo da riuscire a trasmettere concetti di programmazione. Questo libro riduce l'utilizzo di costrutti potenzialmente passibili di errore attraverso l'impiego di funzionalità più attuali dello standard di C++, come i parametri di riferimento, la libreria di stream, la classe `string` e il modello `vector<T>`. I puntatori vengono utilizzati principalmente per il polimorfismo e l'implementazione di liste collegate.
- *Utilizzo immediato degli oggetti.* Gli oggetti vengono introdotti in due fasi. Dal Capitolo 2 in avanti i lettori impareranno a utilizzare gli oggetti, in particolare stringhe, stream, istanze delle classi semplici `Time` ed `Employee` e alcune funzioni grafiche. I lettori acquisiranno familiarità con i concetti di creazione degli oggetti e di chiamata delle funzioni membro man mano che il libro prosegue lungo un percorso tradizionale, trattando il branching e i cicli, le funzioni e le procedure. Nei Capitoli 5 e 6 i lettori impareranno a implementare le classi e le funzioni membro.
- *Utilizzo facoltativo delle funzioni grafiche.* A molti utenti potrebbe interessare l'approfondimento degli aspetti legati alla grafica. Questo libro include numerosi esercizi in cui la programmazione tradizionale e quella grafica si rafforzano a vicenda. Per farlo viene utilizzata una libreria grafica molto semplice che è disponibile per diverse piattaforme molto diffuse. A differenza delle librerie grafiche tradizionali, questa utilizza gli oggetti in modo molto diretto ed efficace. L'utilizzo di questa libreria è

facoltativo. Il Capitolo 8 contiene un'introduzione alla programmazione di interfacce grafiche utente mediante l'impiego di un toolkit open-source (utilizzabile gratuitamente) simile alla libreria Microsoft Foundation Class.

La scelta del linguaggio di programmazione ha un notevole impatto sull'efficacia didattica di qualsiasi libro che affronti l'argomento. Lo scopo di questo libro è tuttavia quello di insegnare concetti informatici e non tutti i dettagli di C++. Questo linguaggio viene utilizzato come strumento per padroneggiare i fondamenti della scienza informatica.

Impostazione pedagogica

Ogni capitolo inizia con una panoramica degli obiettivi trattati e un'introduzione a supporto. I capitoli sono disseminati di cinque tipi di note che aiutano il lettore, chiamate *Errore comune*, *Suggerimento*, *Consiglio*, *Approfondimento* e *Curiosità*. Queste note sono segnalate in modo da non interrompere il flusso del materiale principale. Alcune note sono brevi, altre occupano più pagine e hanno lo scopo di rendere più completa e convincente la spiegazione degli argomenti.

- La nota *Errore comune* descrive gli errori commessi più spesso dagli utenti e spiega perché un errore si verifica e come comportarsi per la sua risoluzione.
- La nota *Suggerimento* illustra alcune pratiche aggiuntive di buona programmazione. Poiché la maggior parte di esse richiede un ulteriore sforzo di apprendimento, queste note motivano con precisione la ragione del suggerimento e spiegano perché lo sforzo verrà ripagato.
- La nota *Consiglio* spiega come utilizzare gli strumenti a disposizione dell'utente in modo più rapido ed efficace. Molti utenti alle prime armi riflettono poco sul proprio modo di utilizzare il computer e il software, e spesso non hanno familiarità con i trucchi del mestiere, come le scelte rapide da tastiera, la ricerca e sostituzione o l'automazione di operazioni comuni mediante gli script.
- La nota *Approfondimento* tratta gli argomenti non essenziali o gli spunti più complessi. Alcuni di questi argomenti introducono costruzioni sintattiche alternative che possono anche non essere tecnicamente avanzate. In molti casi il libro utilizza un costrutto sintattico particolare ma spiega anche le alternative disponibili proprio in queste note. Molti utenti apprezzano questo approccio semplice, perché riduce di molto il numero di decisioni autonome da prendere.
- La nota *Curiosità* fornisce alcune informazioni storiche e sociali sul mondo dei computer, oltre ad alcuni articoli di particolare interesse sui progressi della scienza informatica.

La maggior parte degli esempi è disponibile sotto forma di programmi completi, pronti per essere eseguiti. I programmi sono disponibili in forma elettronica.

L'Appendice A contiene la guida di stile che è stata utilizzata per questo libro. Avere uno stile di programmazione coerente è molto importante, ma ogni lettore è libero di adottare uno stile personale. Anche la guida di stile è disponibile in forma elettronica.

L'Appendice B contiene un riepilogo della sintassi e la documentazione su tutte le funzioni e le classi delle librerie utilizzate in questo libro.

Struttura del libro

- Capitolo 1: Introduzione
- Capitolo 2: Tipi di dati fondamentali
- Capitolo 3: Oggetti
- Capitolo 4: Controllo di base del flusso
- Capitolo 5: Funzioni
- Capitolo 6: Classi
- Capitolo 7: Controllo avanzato del flusso
- Capitolo 8: Collaudo e debugging
- Capitolo 9: Vettori e array

Si noti che la libreria di funzioni grafiche trattata nel Capitolo 3 è facoltativa.

Se si preferisce seguire un percorso che tratti l'ereditarietà e il design orientato agli oggetti, si possono privilegiare i seguenti capitoli.

- Capitolo 10: Puntatori
- Capitolo 11: Ereditarietà
- Capitolo 12: Stream
- Capitolo 13: Design orientato agli oggetti

I capitoli elencati di seguito forniscono un'introduzione agli algoritmi e alle strutture di dati.

- Capitolo 14: Ricorsività
- Capitolo 15: Ordinamento e ricerca
- Capitolo 16: Introduzione alle strutture di dati

I capitoli finali possono essere utilizzati come coronamento del percorso scelto.

- Capitolo 17: C++ avanzato
- Capitolo 18: Interfacce grafiche utente

La figura della pagina successiva mostra le dipendenze tra i capitoli. L'ordine di lettura può seguire i percorsi suggeriti.

Risorse Web

All'indirizzo <http://www.ateneonline.it/horstmann> è reperibile materiale aggiuntivo per studenti e docenti.

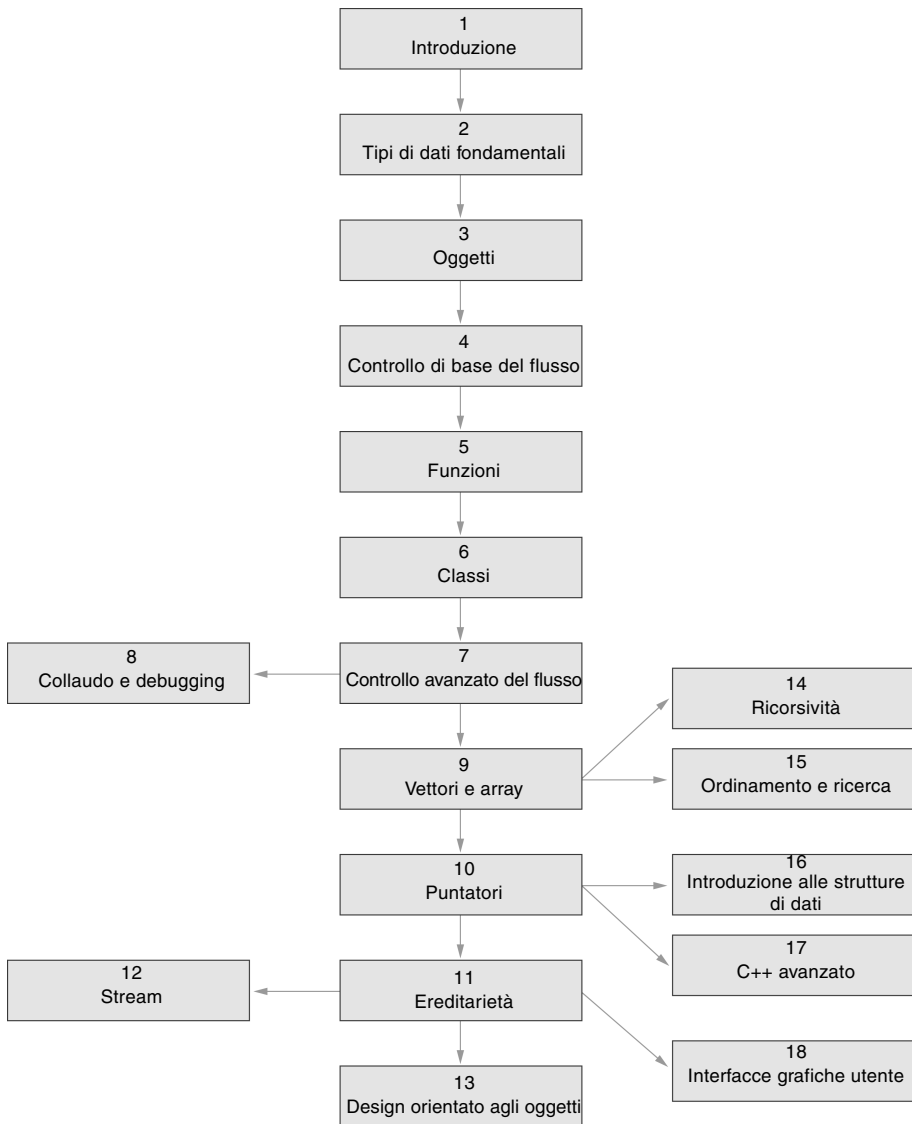


Figura P.1 Dipendenze tra i capitoli.

Ringraziamenti

Si ringraziano i professori Giorgio Satta, Pier Luca Montessoro, Vincenzo Grassi, Andrea Schaerf, Eduardo Calabrese, Giuliano Laccetti per la gentile consulenza che ha contribuito alla decisione della Casa Editrice di pubblicare il volume.