

# Guida rapida all'insegnamento

## Programmazione 2

Docente: Giuseppe Scollo

Università di Catania, sede di Comiso (RG)  
Facoltà di Scienze Matematiche, Fisiche e Naturali  
Corso di Studi in Informatica applicata, AA 2007-8

### Indice

1. Guida rapida all'insegnamento
2. Organizzazione dell'insegnamento
3. Obiettivi formativi
4. Attività formative
5. Relazione con altri insegnamenti
6. Programma delle lezioni (1)
7. Programma delle lezioni (2)
8. Modalità di valutazione
9. Testi consigliati
10. Strumenti per l'interazione formativa
11. Alcuni siti utili

## Organizzazione dell'insegnamento

- **modalità:** blended e-learning
  - proposta: proviamo a usare il Wiki, ad es. per esercizi e progetti?
- sviluppo progettuale cooperativo
  - non solo di progetti di laboratorio, ma anche dello stesso progetto formativo dell'insegnamento
- condivisione di obiettivi formativi
- imparare **metodi** di programmazione, più che un linguaggio

## Obiettivi formativi

Acquisizione e sviluppo della capacità di:

- comprendere le proprietà fondamentali di una vasta gamma di algoritmi, e di strutture dati per essi, di largo impiego nella pratica professionale della programmazione
- realizzare gli algoritmi studiati in un linguaggio orientato agli oggetti, quale il C++, guadagnando familiarità con l'uso di strumenti di supporto che siano:
  - conformi alla definizione standard del linguaggio,
  - affidabili,
  - atti alla generazione di efficienti programmi eseguibili.

## **Attività formative**

L'organizzazione dell'insegnamento prevede 48 ore di lezione (ed esercitazione)

L'acquisizione di metodi e competenze professionali nella disciplina è sostenuta da:

- frequenza delle lezioni ed esercitazioni
- studio del testo [Sedgewick, 2003]
- elaborazione di soluzioni a problemi ed esercizi proposti
- consultazione di altri testi e materiali didattici
- interazione con il docente (ricevimento settimanale)
- collaborazione con i colleghi

## **Relazione con altri insegnamenti**

Quale bagaglio preliminare è

- raccomandato
  - significa: nessun vincolo di propedeuticità sugli esami, ma seguire i due insegnamenti nella sequenza naturale è ottimale per il conseguimento degli obiettivi formativi

l'insegnamento di Programmazione 1

## **Programma delle lezioni (1)**

1. Introduzione ad algoritmi e strutture dati in C++
2. Principi di progettazione degli algoritmi
3. Elementi di analisi degli algoritmi
4. Approssimazione asintotica di funzioni
5. Ricorrenze e stima delle prestazioni
6. Tipi di dati elementari
7. Costrutti di base per strutture dati
8. Liste concatenate
9. Elaborazione elementare delle liste
10. Allocazione di memoria per le liste
11. Stringhe e array multidimensionali
12. Liste multiple e rappresentazione di grafi

## **Programma delle lezioni (2)**

13. Algoritmi e strutture dati per il Problema  $3x+1$
14. Riepilogo di fondamenti e strutture dati
15. Tipi di dati astratti
16. Sviluppo di ADT generici
17. Ereditarietà e polimorfismo, ADT code
18. Code generalizzate, ADT di prima categoria
19. Ricorsione e induzione
20. Algoritmi ricorsivi
21. Programmazione dinamica
22. Alberi, alberi binari
23. Algoritmi ricorsivi su alberi binari e su grafi
24. Riepilogo di ADT e ricorsione

## Modalità di valutazione

### prova scritta

alcuni quesiti a risposta chiusa (30-40%)

qualche quesito a risposta aperta (con limiti di spazio) (10-20%)

soprattutto: esercizi di programmazione (50%)

### eventuale esame orale (opzionale)

colloquio individuale sugli argomenti del programma, a partire dal lavoro svolto dallo studente in un progetto, meglio se realizzato in collaborazione con altri studenti

### valutazione in itinere

anticipazioni della prova scritta

contributi prodotti dagli studenti in risposta a problemi ed esercizi

proposti

=> **bonus!**

Il superamento dell'esame porta all'acquisizione di 6 crediti.

## Testi consigliati

### Testo di riferimento (adottato)

R. Sedgewick

Algoritmi in C++, Terza Edizione: Fondamenti, strutture dati, ordinamento, ricerca  
Addison-Wesley, Pearson Education Italia (2003)

### Altri testi consigliati (per consultazione)

L. Joyanes Aguilar, Fondamenti di programmazione in C++, McGraw-Hill (2008)

H.M. Deitel, P.J. Deitel, C++, Fondamenti di programmazione, Seconda Edizione, Prentice-Hall, Apogeo (2005)

H.M. Deitel, P.J. Deitel, C++, Tecniche avanzate di programmazione, Seconda Edizione, Prentice-Hall, Apogeo (2006)

M.T. Goodrich, R. Tamassia, D.M. Mount, Data Structures and Algorithms in C++, John Wiley & Sons (2004)

B. Stroustrup, The C++ Programming Language, Third Edition, Addison-Wesley (2004)

## Strumenti per l'interazione formativa

Forum e Wiki: cosa va dove?

- Forum: discussioni di
  - organizzazione dell'insegnamento
  - argomenti delle lezioni
  - problemi con strumenti di laboratorio
- Wiki: è incoraggiata la documentazione in corso d'opera dello sviluppo di esercizi, programmi e progetti, facendo pieno uso della funzionalità Wiki per il controllo di versione e per la gestione di discussioni

## Alcuni siti utili

### Testi liberamente disponibili

Bjarne Stroustrup's **C++ Glossary**:

<http://public.research.att.com/~bs/glossary.html>

Free electronic book by Bruce Eckel (2 voll): **Thinking in C++, 2nd Ed.:**

<http://mindview.net/Books/TICPP/ThinkingInCPP2e.html>

Doug Schmidt's **C++ Tutorials**: <http://www.cs.wustl.edu/~schmidt/C++>

### Strumenti open-source

**GCC, the GNU Compiler Collection**: <http://gcc.gnu.org>