

Programmazione II

Laurea Triennale
in Informatica

Docenti:

Prof.ssa Federica Mandreoli (gruppo A-L)

e-mail: federica.mandreoli@unimore.it

Prof. Luca Bedogni (gruppo M-Z)

e-mail: luca.bedogni@unimore.it

Obiettivi dell'insegnamento

Fornire **competenze e abilità avanzate** nella programmazione a partire dalle conoscenze di base acquisite durante il corso di Programmazione I

Implementare e sperimentare **strutture dati** e **algoritmi** in parallelo al corso di Algoritmi e strutture dati

Sviluppare **capacità di sviluppo e testing** di progetti scritti in linguaggio C++

Da dove partiamo?

- Partiamo dalle conoscenze acquisite nel corso di **Programmazione I**
- Non è necessario aver superato l'esame di Programmazione I per seguire Programmazione II
...anzi il consiglio è di *seguire* comunque il corso di Programmazione II anche se non avete superato Programmazione I e di *provare sempre* a svolgere gli esercizi

Programma delle lezioni: implementazione di strutture dati dinamiche e algoritmi

- Puntatori: Ripasso e nuovi concetti
- ADT e strutture dati dinamiche
- Implementazione di liste

- Implementazione di algoritmi ricorsivi
- Gestione chiamate e stack
- Implementazione di soluzioni ricorsive a problemi notevoli

- Implementazione di alberi binari di ricerca
- Implementazione di alberi n-ari
- Visite di alberi

- Implementazione di grafi
- Visite di grafi
- Ricerca dei cammini minimi

Liste

Ricorsione

Alberi

Grafi

**Liste, Pile e
code**

Ricorsione

**Alberi &
Grafì**

Algoritmi e strutture dati

Programma delle lezioni: programmazione C++

Bit vector

**Funzioni di
ordine
superiore**

Programma delle lezioni: sviluppo e testing

Debugging

- definizione di bug
- attività di debugging
- debugging in un IDE

Moduli file

- modularizzazione di un progetto
- suddivisione in file
- header file
- visibilità e scoping

Compilazione separata

- fasi di compilazione
- linking statico e dinamico
- makefile

Documentazione

- Principi di documentazione del codice
- Doxygen

Version control

- controllo di versioni di un progetto
- gitLab

Metodologia

- Nel corso si mantiene l'approccio "imparare facendo" (**hands on**) visto in Programmazione I:
 - Ogni nuova nozione o dettaglio viene messo in pratica al computer durante la lezione
 - Esercizi di complessità crescente
- Questo approccio ha dei **pro** e dei **contro**

Repetita iuvant



Introduzione al corso di
Programmazione I

- Tutte le lezioni si svolgono in laboratorio
 - In laboratorio potete usare il vostro portatile
 - Consigliamo di usare anche i PC del laboratorio

Testi di riferimento

Luis Joyanes Aguilar.

«Fondamenti di Programmazione in C++ II edizione»

McGraw-Hill, 2021 (Testo per adottato per il corso)

P. Foggia, M. Vento.

«Algoritmi e strutture dati - Astrazione, progetto e realizzazione»,

McGraw-Hill, 2011 (Testo per approfondimenti)

B. Eckel. **«Thinking in C++»**, 2nd Edition, Volume 1 - Sito per il download gratuito: <http://www.mindviewinc.com/> (già adottato nel corso di «Programmazione I»)

Ambiente di programmazione

- Useremo l'ambiente VISUAL STUDIO CODE (VS CODE)
- VS CODE è installato sulle macchine del laboratorio base e potete installarlo sui vostri computer
- VS CODE facilita la realizzazione di programmi anche complessi in C++ grazie ad una serie di tool per la scrittura di codice in C++, per il debugging, per la creazione di progetti, ...
- VS CODE è un'interfaccia e utilizza il compilatore C++ (g++) per l'esecuzione delle operazioni

**PER LA PRIMA LEZIONE:
INSTALLATE VS CODE E PROVATE A LANCIARE IL
DEBUGGER!!**

Sito del corso

Sito del corso su **Moodle**

- Informazioni sul corso
- Materiale didattico
- Notifiche tramite gli Annunci (ad esempio in caso di annullamento della lezione, materiale da scaricare, pubblicazione risultati appelli,...)

Modalità d'esame

- Obbligo di iscrizione su ESSE3
- **Prova Intermedia (teoria, circa a meta Aprile)**
 - Test con domande a risposta multipla e domande a risposta aperta
 - Valido per i primi 2 appelli estivi
- **Prova scritta (teoria)**
 - test con domande a risposta multipla e domande a risposta aperta
- **Prova di programmazione (pratica)**
 - La prova si tiene in Laboratorio Base
 - implementazione al calcolatore di un programma in un tempo limitato
- Voto: 65% prova di programmazione - 35% prova scritta
- **Orale facoltativo**
- Ripareremo delle regole per la partecipazione agli appelli nel proseguo del corso

Tutorato

- **20 ore di tutorato** ovvero supporto allo studio della materia da parte di uno studente della laurea magistrale
RICCARDO MESCOLI
- Le ore di tutorato si svolgeranno in Laboratorio per consentirvi di prendere dimestichezza con le macchine su cui si svolgerà la prova di laboratorio
- Durante le lezioni vi lascerò degli esercizi da svolgere
- Durante le ore di tutorato potrete svolgere gli esercizi assieme al tutor o discutere assieme a lui le vostre soluzioni...
- Il tutorato inizierà la 3° settimana

Quanto è difficile programmazione II?

Statistiche AA 2017-2018

- #iscritti I anno CdL Informatica AA 2017-2018: 184
- #studenti che hanno superato Programmazione II: 45
- Media voto: 23,5

Statistiche AA 2018-2019

- #iscritti I anno CdL Informatica AA 2018-2019: 195
- #studenti che hanno superato Programmazione II: 49
- Media voto: 23,1

Statistiche AA 2019-2020

- #iscritti I anno CdL Informatica AA 2019-2020: 194
- #studenti che hanno superato Programmazione II: 47
- Media voto: 23,2

Statistiche AA 2020-2021

- #iscritti I anno CdL Informatica AA 2020-2021: 160
- #studenti che hanno superato Programmazione II: 61
- Media voto: 25

Riassumendo ...

The bad news 😞 Solo il 38% degli studenti iscritti al 2° anno che attualmente seguono il secondo periodo hanno superato l'esame

The good news 😊 Quasi la totalità degli studenti che hanno seguito costantemente e con attenzione fino alla fine del corso hanno già superato l'esame

Alcuni consigli

- Imparare a ragionare sulla struttura di un programma prima di mettersi a scrivere codice
- Fate domande durante la lezione se non avete capito!!
- Venite a ricevimento per risolvere dubbi e comprendere gli errori commessi!!

