



**UNIVERSITÀ
DI PARMA**

**Informatica e Laboratorio di
Programmazione**
Alberto Ferrari

- Informatica e laboratorio di programmazione
- **crediti:**
 - **9 CFU** (corrispondenti a circa 72 ore di lezione)
- **orario:**
 - Lunedì 13.30 - 15.30 aula 7
 - Lunedì 15.30 – 17.30 lab. Informatica 1,2,3
 - Mercoledì 8.30-10.30 – aula 7
- **docente:** Prof. Alberto Ferrari (alberto.ferrari@unipr.it)
 - Phone: +39 0521 90 5708
 - ricev.: Ingegneria Sede Scientifica – Palazzina 1 – lunedì 12:30-13:30
(*solo su appuntamento – alberto.ferrari@unipr.it*)
- **materiale:** <https://albertoferrari.github.io/>

- introduzione alla programmazione
 - algoritmo – risolutore - esecutore
 - programmazione strutturata
 - collezioni e flussi di dati
 - funzioni e ricorsione
 - oggetti e astrazioni
 - interfacce grafiche
- introduzione all'informatica
 - rappresentazione dei dati
 - concetto di computazione
 - sistemi di elaborazione
 - sviluppo del software

- rappresentazione dei dati
 - numeri – testi – immagini - suoni
- teoria della computazione
 - linguaggi - automi e calcolabilità - complessità
- sviluppo del software
 - ciclo di vita - qualità e collaudo
- sistemi di elaborazione
 - calcolatori - sistemi operativi

- algoritmi in Python 3
- collezioni e flussi di dati
- funzioni
- oggetti e interfacce
- linguaggio C++11
- interfacce grafiche

- lezioni in aula
- soluzione guidata di esercizi in aula
- esercizi di programmazione in laboratorio
 - le esercitazioni in laboratorio sono centrali per il corso
 - gli esercizi proposti vertono sugli stessi argomenti generali delle lezioni in aula

- esercizi di base, in *Python*
- programmazione ad oggetti e semplici progetti
 - prima in *Python* e poi in *C++*
- ***attenzione: non sono sufficienti le poche ore in laboratorio per imparare a programmare!***
 - ripassare i contenuti
 - completate ***tutti*** gli esercizi, a casa

If you wish to learn swimming you have to go into the water and if you wish to become a problem solver you have to solve problems. (George Polya)

- verificare in anticipo il proprio account per l'accesso ai lab
 - <http://www.cedi.unipr.it/gestioneaccounts>
- possibile programmare (esercitazioni e progetti) ***da soli*** o ***in coppia***
 - tassativamente non più di due
- verifica esercitazioni
- al termine di ogni esercitazione, ***consegna*** dei programmi in una cartella denominata con la propria ***matricola***
- periodicamente, lavori consegnati sottoposti a ***valutazioni intermedie***, non preannunciate

- libri di testo e *slide* del corso
- esercizi risolti
- durante le esercitazioni, assistenza di *tutor*
- a lezione
 - esempi prima delle esercitazioni
 - soluzioni di esercizi delle settimane precedenti

- l'esame consiste in
 - una prova sui concetti teorici (quiz)
 - calendario appelli
 - una prova di programmazione (lo sviluppo di un'applicazione software da sviluppare in laboratorio)
 - è previsto l'utilizzo di prove di valutazione in itinere
 - prova in appelli usuali (~3 ore)
 - le due prove si possono svolgere in tempi diversi e ciascuna resta valida per l'intero anno (fino ad ottobre)
- il ***voto finale*** è determinato:
 - per 1/4 dalla teoria
 - per 3/4 dalla programmazione

- ***Informatica e Laboratorio di Programmazione (A.A. 2019-2020).***
McGraw-Hill custom publishing. ISBN 978-13-074-5700-1 (27€, nelle librerie universitarie)
- ***A.B. Downey et al.: How to Think Like a Computer Scientist: Learning with Python 3, 3rd Edition***
- *- <http://openbookproject.net/thinkcs/>*
- ***M. Beri: Python, Apogeo Pocket, 2010, 978-8850329151 (~8€)***
- ***A.B. Downey: How to Think Like a Computer Scientist C++ Version***
- <http://greenteapress.com/thinkcpp/>

- C. S. Horstmann: Fondamenti di C++, McGraw-Hill 2003, 978-8838661051 (~43€, 768pp.)
- S.B. Lippman, J. Lajoie, B.E. Moo: C++ Primer 5e, Addison Wesley 2012, 978-0321714114 (~36€, 940pp.)
- M. Dawson: Python Programming for the Absolute Beginner 3e, Course Tech. 2010, 978-1435455009 (~25€, 450pp.)
- M. Dawson: Beginning C++ Through Game Programming 4e, Course Tech. 2014, 978-1305109919 (~28€, 390pp.)
- A. Lorenzi, V. Moriggia: Programmazione ad oggetti e linguaggio C++, Atlas 2004, 978-8826811956 (~12€, 380pp.)
- Oppure altro testo di programmazione orientata agli oggetti in C++ per le scuole superiori
- MIT: Intro to CS and Programming in Python - Video delle lezioni (<https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-0001-introduction-to-computer-science-and-programming-in-python-fall-2016/lecture-videos/index.htm>)

- Strumenti *open source, multi-piattaforma*
- **Python 3**, battery included: <http://python.org/>
 - PyGame: <http://pygame.org/>
 - Brython: <http://brython.info/>
 - Brython playground: <http://www.ce.unipr.it/brython/>
- **C++11**: <http://isocpp.org/>
 - documentazione
 - <http://cplusplus.com/reference/>
 - <http://cppreference.com/>
 - MinGV (compilatore e librerie)
 - <http://www.mingw.org/>

