



Pattern Recognition  
and Applications Lab

# Linguaggi di Programmazione Orientati agli Oggetti

**Docente:** Ambra Demontis

**Anno Accademico:** 2023 - 2024

Corso di Laurea in Ingegneria Elettrica, Elettronica e Informatica



University of Cagliari,  
Italy

Department of Electrical and  
Electronic Engineering



# I Vantaggi della Programmazione Orientata agli Oggetti

- **Modularità:** Il codice è altamente **strutturato** (organizzato), quindi (1) è **facile da estendere**; (2) è **facile localizzare** eventuali **problemi**.
- **Riutilizzo di codice esistente:** Permette di riutilizzare lo stesso codice per trattare casi simili ma con alcune caratteristiche differenti.

**NB: Vantaggi utilissimi quando si implementano programmi complessi**

In questo corso utilizzeremo il linguaggio “Python”.

# Perchè Python

## Vantaggi:

- Facile da imparare
- Creare programmi complessi richiede scrivere poco codice
- Tante librerie a disposizione (machine learning, servizi web etc.)
- Comunità di sviluppatori molto attiva

## Svantaggi:

- Meno computazionalmente efficiente di altri linguaggi e.g. C
- Tuttavia: (1) esistono librerie che traducono automaticamente il codice in C  
(2) diverse librerie per Python permettono di sfruttare la GPU

# Prerequisiti

La conoscenza di almeno un linguaggio di programmazione procedurale-imperativo, ad esempio C o Pascal.

Quanti linguaggi di programmazione conoscete?

Avete già programmato in Python?

# Propedeuticità

Per poter sostenere l'esame di linguaggi di programmazione orientati agli oggetti è necessario aver superato gli esami di:

- Analisi matematica 2
- Fisica 2

Per chi non li avesse ancora superati è altamente consigliato seguire comunque le lezioni.

# Programma (in Macro-Blocchi)

## Prima Parte:

- La programmazione orientata agli oggetti
- Metodi di rappresentazione degli oggetti (diagrammi di classe)
- Le basi del linguaggio Python

## Seconda Parte:

- Utilizzo avanzato del linguaggio Python e della programmazione orientata agli oggetti.

# Cosa Imparerete

Dal punto di vista teorico:

- Concetti avanzati riguardo la programmazione orientata agli oggetti (OOP)

Dal punto di vista pratico:

- Creare il prototipo di un programma utilizzando i diagrammi di classe
- Creare programmi di media complessità in Python utilizzando OOP

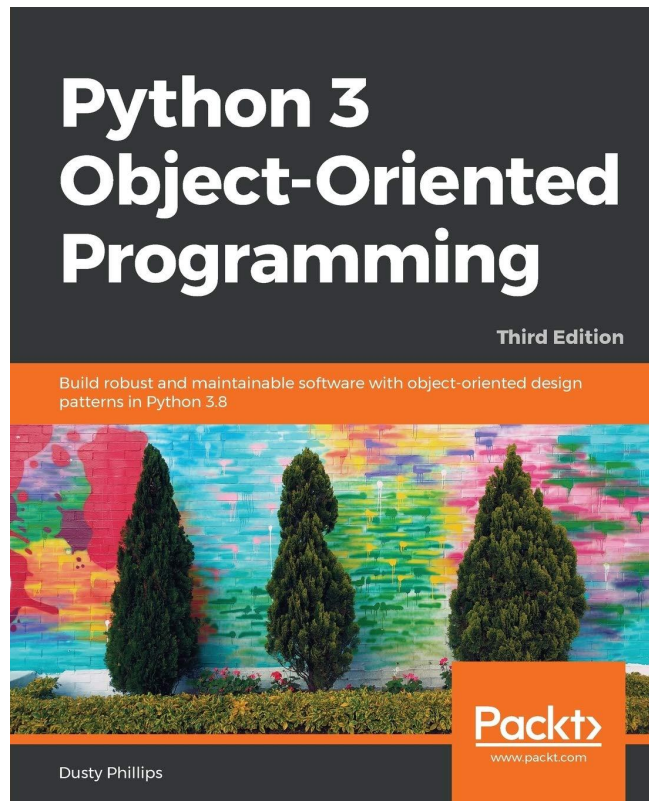
# Testo di Riferimento

Python 3 Object-Oriented Programming:  
Build robust and maintainable software  
with object-oriented design patterns in  
Python 3.8

**Autore:** Dusty Phillips

**Editore:** packt

**Edizione:** III





# Esame

Esame scritto composto da:

- quesiti di teoria riguardanti la programmazione orientata agli oggetti
- esercizi da svolgere utilizzando il linguaggio di programmazione Python.

Votazione massima 33/30.

Quiz a risposta multipla a metà corso con votazione massima due punti bonus sul voto d'esame.

Verrà effettuata una simulazione d'esame a fine corso.

**NB: Il corso (e quindi l'esame è sui linguaggi di programmazione orientati agli oggetti, non su Python...)**

# Software e Materiale Didattico

## Ambiente di sviluppo

PyCharm (IDE) with Python 3

<https://www.jetbrains.com/pycharm/>

**NB: si raccomanda di installarlo entro questa settimana.**

## Materiale didattico

Troverete tutto il materiale didattico nel sito web del corso:

<https://unica-lpo.github.io/>

# Comunicazioni

Ricevimento previo appuntamento.

Inoltrare una mail all'indirizzo: [ambra.demontis@unica.it](mailto:ambra.demontis@unica.it)

Oggetto: LPO-2023-2024 ricevimento

Google group del corso: LPO-2023-2024