



# UNIVERSITÀ DI PISA

---

## PARADIGMI DI PROGRAMMAZIONE

### FRANCESCO GAVAZZO

Anno accademico	2022/23
CdS	INFORMATICA
Codice	733AA
CFU	9

Moduli	Settore/i	Tipo	Ore	Docente/i
PARADIGMI DI PROGRAMMAZIONE	INF/01	LEZIONI	72	FABIO GADDUCCI FRANCESCO GAVAZZO

#### Obiettivi di apprendimento

##### *Conoscenze*

Imparare i principi alla base dei principali paradigmi di programmazione (funzionale, object-oriented e concorrente) e del funzionamento dei linguaggi di programmazione moderni.

##### *Modalità di verifica delle conoscenze*

Esame finale, scritto e orale.

##### *Capacità*

Imparare a ragionare sul funzionamento dei linguaggi di programmazione in astratto, tramite modelli formali del loro comportamento. Imparare ad apprendere nuovi linguaggi di programmazione. Imparare a programmare nel linguaggio OCaml.

##### *Modalità di verifica delle capacità*

Esercitazioni durante il corso ed esame finale.

##### *Comportamenti*

Programmare in modo consapevole, avendo cognizione di cosa accade al programma in fase di compilazione ed esecuzione.

##### *Modalità di verifica dei comportamenti*

Esercitazioni durante il corso ed esame finale.

##### *Prerequisiti (conoscenze iniziali)*

Elementi di linguaggi formali, semantica e programmazione visti nei corsi del primo anno. Conoscenza del linguaggio JavaScript.

##### *Indicazioni metodologiche*

Lezioni frontali con slide e uso dei jupyter notebook in OCaml.

##### *Programma (contenuti dell'insegnamento)*

1. *Paradigma di programmazione funzionale.* L'essenza: il  $\lambda$ -calcolo; il linguaggio OCaml; formalizzazione di un linguaggio funzionale; implementazione in OCaml di un interprete del linguaggio funzionale; sistemi di tipi nei linguaggi funzionali.
2. *Paradigma di programmazione object-oriented.* L'astrazione dei dati e gli oggetti; modularità, incapsulamento, ereditarietà, principio di sostituzione, e class-based vs object-based; formalizzazione di linguaggi object-oriented; Sistemi di tipi nei linguaggi object-oriented; tecniche di implementazione di linguaggi class-based (es. Java Virtual Machine); Esempi di implementazione in OCaml.
3. *Paradigma di programmazione concorrente.* Concetti di base: esecuzione non sequenziale; formalizzazione di un modello di concorrenza; costrutti di programmazione concorrente nei linguaggi moderni.



## UNIVERSITÀ DI PISA

---

### Bibliografia e materiale didattico

#### Materiale didattico

- slide e dispense fornite dal docente
- Jupyter notebook in OCaml

#### Libri di testo consigliati (per approfondimenti)

- Real World OCaml (<https://dev.realworldocaml.org/>)
- P. Sestoft, Programming Language Concepts, Springer, 2017.
- M. Gabrielli e S. Martini, Linguaggi di programmazione: Principi e Paradigmi, McGraw-Hill, 2006.

### Modalità d'esame

Scritto e orale.

*Ultimo aggiornamento 02/10/2022 22:55*