

PROGRAMMAZIONE II

GIAN-LUIGI FERRARI

Anno accademico	2019/20
CdS	INFORMATICA
Codice	273AA
CFU	9

Moduli	Settore	Tipo	Ore	Docente/i
PROGRAMMAZIONE II	INF/01	LEZIONI	72	GIAN-LUIGI FERRARI

Obiettivi di apprendimento

Conoscenze

Al termine dell'insegnamento:

- lo studente avrà acquisito conoscenze in merito alle tecniche e alle metodologie di base per la progettazione e realizzazione di semplici applicazioni nel paradigma a oggetti.
- lo studente avrà acquisire conoscenze rispetto alla progettazione e realizzazione dei linguaggi di programmazione

Modalità di verifica delle conoscenze

Per l'accertamento delle conoscenze saranno svolte delle prove in itinere (prove scritte) e progetti di programmazione

Capacità

Al termine del corso:

- lo studente saprà utilizzare il linguaggio Java per la progettazione e realizzazione di applicazioni nel paradigma object oriented
- lo studente avrà le conoscenze per comprendere quali sono le tecniche per la realizzazione dei linguaggi di programmazione,
- lo studente avrà utilizzato il linguaggio OCAML per simulare la struttura del run-time dei linguaggi di programmazione

Modalità di verifica delle capacità

- Durante le attività di laboratorio saranno svolte attività pratiche di programmazione
- Lo studente dovrà preparare e presentare una relazione scritta che riporti i risultati delle attività di progetto

Comportamenti

Abilità nel lavoro di gruppo (team work)

Modalità di verifica dei comportamenti

Verrà valutata la correttezza e la documentazione delle attività (progetti) di programmazione.

Prerequisiti (conoscenze iniziali)

Conoscenza della programmazione imperativa e delle tecniche algoritmiche di base.

Indicazioni metodologiche

La frequenza non è obbligatoria ma è *fortemente consigliata*. Il corso è pensato ed organizzato per una frequenza attiva e continua.

Programma (contenuti dell'insegnamento)

Il corso si articola attorno a due argomenti principali.

- **Programmazione orientata ad oggetti.** Il nucleo del linguaggio Java: classi, oggetti, ereditarietà, astrazioni sui dati (tecniche di progettazione, specifica e implementazione, tecniche di verifica), astrazione mediante gerarchie, polimorfismo in Java. Tecniche

di programmazione concorrente (cenni).

- **Paradigmi dei linguaggi di programmazione.** Strutture a run time: ambiente, memoria, heap; blocchi, sottoprogrammi, records di attivazione, pila dei records di attivazione; oggetti; interpretazione, compilazione. Progettazione di interpreti in Ocaml.

Bibliografia e materiale didattico

Per la parte relativa alle tecniche di programmazione orientata agli oggetti il testo di riferimento è:

- Object-Oriented: B. Liskov (with J. Guttag), Program Development in Java: Abstraction, +Specification and Object-Oriented design, Addison-Wesley, 2000

Note didattiche per la programmazione in Java: Programming Languages and Techniques, Zdancewic, Pierce, Weirich (2018) [download](#).

Per la parte relativa ai paradigmi di programmazione il testo di riferimento:

- M. Gabrielli and S. Martini, Linguaggi di Programmazione: Principi e Paradigmi, McGraw-Hill, 2006

Per la parte relativa alla programmazione di interpreti in OCAML si consulti:

- Note on Programming Language Concepts [Note \(pdf\)](#).

Ulteriore materiale didattico è indicato nella descrizione degli argomenti trattati a lezione

Modalità d'esame

- L'esame è composto da una prova scritta e una discussione orale di un progetto di programmazione
- La prova scritta consiste in una serie di esercizi da risolvere (durata due ore)

Altri riferimenti web

Sono presenti nella pagina web del corso.

Ultimo aggiornamento 04/09/2019 09:02