



# UNIVERSITÀ DEGLI STUDI DI ENNA "KORE"

Facoltà di Ingegneria e Architettura

Anno Accademico 2022/2023

Corso di studi in Ingegneria Informatica, classe di laurea L8

Insegnamento	Linguaggi di Programmazione ad Oggetti
CFU	9
Settore Scientifico Disciplinare	ING-INF/05
Nr. ore di aula	54
Nr. ore di studio autonomo	171
Nr. ore di laboratorio	0
Mutuazione	Nessuna
Annualità	II
Periodo di svolgimento	Secondo Semestre

Docente	E-mail	Ruolo <sup>i</sup>	SSD docente
SORCE Salvatore	<a href="mailto:salvatore.sorce@unikore.it">salvatore.sorce@unikore.it</a>	PA	ING-INF/05

Propedeuticità	Fondamenti di Informatica
Prerequisiti	Sono richieste le conoscenze di base della programmazione strutturata
Sede delle lezioni	Facoltà di Ingegneria e Architettura

## Moduli

N.	Nome del modulo	Docente	Durata in ore

## Orario delle lezioni

L'orario delle lezioni sarà pubblicato sull'Agenda WEB dell'Università degli Studi di Enna "Kore"

[https://gestioneaule.unikore.it/agendaweb\\_unikore/](https://gestioneaule.unikore.it/agendaweb_unikore/)

## Obiettivi formativi

Lo scopo dell'insegnamento è fornire allo studente le nozioni fondamentali relative ai linguaggi di programmazione ad alto livello e un buon livello di conoscenza di diversi paradigmi di programmazione (in particolare object-oriented/orientato agli oggetti). Verranno presentati il paradigma orientato agli oggetti e le relative metodologie di programmazione, usando per le rispettive implementazioni il linguaggio Java e i class diagram dell'UML (Unified Modeling Language).

## Contenuti del Programma

N.	ARGOMENTO	TIPOLOGIA	DURATA
1	Principi dei linguaggi di programmazione. Paradigmi di programmazione (e.g., funzionale, procedurale, logico, etc.). Il paradigma orientato agli oggetti	Frontale	1h
2	Introduzione agli aspetti di base riguardanti la sintassi, la semantica e la notazione dei linguaggi di programmazione.	Frontale	1h
3	Principi di sviluppo dei linguaggi di programmazione:	Frontale	2h

	espressività, leggibilità, costo (per imparare un linguaggio), robustezza. Caratteristiche intrinseche dei linguaggi: modularità, flessibilità, generalità, efficienza, coerenza. Caratteristiche esterne: diffusione, standardizzazione, portabilità, integrabilità.		
4	Introduzione alla programmazione ad oggetti	Frontale	3h
5	Introduzione all'ambiente di sviluppo e alla compilazione in Java.	Frontale	1h
6	Dati numerici in Java (teoria + esempi)	Frontale	2h
7	Istruzioni di selezione e ripetizione in Java (teoria + esempi)	Frontale	2h
8	Classi definite dall'utente in Java (teoria + esempi)	Frontale	5h
9	Gestione delle eccezioni in Java (teoria + esempi)	Frontale	3h
10	Caratteri e Stringhe in Java: Le classi StringBuffer e StringBuilder (teoria ed esempi)	Frontale	5h
11	Array e Collezioni in Java (teoria + esempi)	Frontale	5h
12	Input e Output su file in Java (teoria + esempi)	Frontale	5h
13	Ereditarietà e Polimorfismo in Java (teoria + esempi)	Frontale	5h
14	Programmazione di rete in Java: protocolli Internet, comunicazione client-server, socket	Frontale	5h
15	Tipi generici e contenitori in Java (teoria + esempi)	Frontale	3h
16	Introduzione all'UML e suo uso per la modellazione	Frontale	3h
17	Sviluppo e implementazione di programmi in Java con uso di collezioni e tipi generici	Frontale	2h
18	Sviluppo e implementazione di programmi in Java con gestione dell'ereditarietà e polimorfismo	Frontale	1h

#### Risultati di apprendimento (descrittori di Dublino)

Gli studenti dovranno aver conseguito le seguenti abilità, conoscenze e competenze:

**Conoscenza e capacità di comprensione:** L'insegnamento si propone di completare la formazione di base necessaria per la progettazione sistematica e strutturata di un progetto software, con particolare attenzione alla parte di progettazione orientata agli oggetti.

**Conoscenza e capacità di comprensione applicate:** Alla fine del percorso formativo, lo studente avrà acquisito una buona conoscenza dei principali principi per la progettazione dei linguaggi di programmazione. Inoltre, lo studente sarà in grado di analizzare e comprendere gli elementi distintivi dei linguaggi di programmazione orientati agli oggetti. Lo studente sarà quindi in grado di (i) progettare programmi software secondo il paradigma di progettazione orientato agli oggetti e (ii) implementare il progetto stesso in Java.

**Autonomia di giudizio:** Lo studente sarà in grado di valutare la qualità di una soluzione in termini di semplicità, leggibilità, efficienza e possibilità di riutilizzo. L'autonomia di giudizio sarà valutata esaminando le soluzioni proposte dagli studenti a problemi di media complessità per la gestione dei dati.

**Abilità comunicative:** Lo studente acquisirà la capacità di comunicare e descrivere problematiche inerenti agli argomenti affrontati utilizzando una terminologia appropriata e corretta. Lo studente sarà anche in grado di descrivere il problema utilizzando i class diagram del linguaggio UML.

**Capacità di apprendere:** Lo studente acquisirà la capacità per apprendere i processi di analisi dei requisiti di una proposta progettuale. Inoltre avrà gli strumenti per approfondire autonomamente le conoscenze di base acquisite durante le lezioni.

---

## Testi per lo studio della disciplina

---

### Testi principali:

- Il Linguaggio Java 4/Ed. Manuale ufficiale, K. Arnold - J. Gosling - D. Holmes. Pearson-Addison Wesley, 2006. ISBN: 9788871922768.
- Java – Fondamenti di programmazione, C. Thomas Wu, McGraw-Hill Italia, 2009.

### Testi di approfondimento:

- Linguaggi di programmazione: principi e paradigmi. Maurizio Gabrielli, Simone Martini. McGraw-Hill Italia, 2011

### Materiale didattico a disposizione degli studenti:

- Lucidi messi a disposizione dal docente relativi ai principi di programmazione, esercizi svolti, esempi di progettazione software in Java.

---

## Metodi e strumenti per la didattica

---

Il docente utilizzerà lezioni frontali, durante le quali verranno presentati i concetti legati alla teoria della programmazione ad oggetti e della modellazione progettuale, e mostrati esempi pratici di implementazione ed uso dei relativi costrutti in Java e UML.

---

## Modalità di accertamento delle competenze

---

L'accertamento delle competenze avverrà attraverso un colloquio strutturato nella seguente maniera:

- 1) discussione sulla progettazione e messa a punto, anche in gruppo, di un progetto in Java (si deve prevedere la creazione di un modello UML-like di tutte le classi presenti nel progetto e documentazione in stile JAVA). L'argomento del progetto è assegnato dal docente;
- 2) domande relative alla teoria dei linguaggi di programmazione.

L'incidenza percentuale delle due componenti per la costruzione del voto finale è così ripartita:

- Progetto 70%
- Teoria 40%

La somma supera il 100% per consentire di compensare un risultato non ottimo in una parte con un risultato ottimo in un'altra parte in modo da poter comunque raggiungere la votazione di 30/30 anche con un errore marginale, e nello stesso tempo consente al docente di poter assegnare la lode nel caso di raggiungimento ottimo in entrambe le parti dell'esame.

Il voto sarà espresso, secondo il seguente schema di valutazione:

- Ottimo (30-30 e lode): Ottima conoscenza e comprensione degli argomenti riguardanti la programmazione JAVA (individuazione di una soluzione eccellente per la messa a punto del progetto assegnato) e della teoria dei linguaggi. Eccellenti capacità espositive.
- Molto buono (26-29): Buona conoscenza e comprensione degli argomenti riguardanti la programmazione JAVA (individuazione di una soluzione ottima per la messa a punto del progetto assegnato) e della teoria dei linguaggi. Ottime capacità espositive.
- Buono (24-25): Buona conoscenza e comprensione degli argomenti riguardanti la programmazione JAVA (individuazione di una soluzione buona per la messa a punto del progetto assegnato) e della teoria dei linguaggi. Buone capacità espositive.
- Discreto (21-23): Discreta conoscenza e comprensione degli argomenti riguardanti la programmazione JAVA (individuazione di una soluzione discreta per la messa a punto del progetto assegnato) e della teoria dei linguaggi. Discrete capacità espositive.
- Sufficiente (18-20): Conoscenza minima degli argomenti riguardanti la programmazione JAVA (individuazione di una soluzione accettabile per la messa a punto del progetto assegnato) e della teoria dei linguaggi.
- Insufficiente: Manca di una conoscenza accettabile degli argomenti riguardanti la programmazione JAVA e della teoria dei linguaggi.

### AVVERTENZE:

- La composizione dei gruppi di lavoro è di TRE componenti al massimo.
- L'accertamento delle competenze avverrà attraverso una prova orale sia per la discussione del progetto che per la teoria.

- La discussione del progetto prevede una presentazione di circa 10 minuti per ogni componente del gruppo, così un gruppo formato da tre persone dovrà svolgere una presentazione della durata di circa 30 minuti.
- Dopo un'introduzione sul progetto, la presentazione dovrà mostrare la soluzione evidenziandone gli aspetti peculiari, usando sistemi tipo powerpoint o equivalenti.
- Tutto il materiale relativo all'esame (presentazione, codice sorgente finale della soluzione, eventuali sorgenti dati e librerie esterne) va consegnato in formato digitale al docente non più tardi di una settimana prima della data dell'appello scelto.

---

#### Date di esame

Le date di esami saranno pubblicate sull'Agenda WEB dell'Università degli Studi di Enna "Kore"

[https://gestioneaule.unikore.it/agendaweb\\_unikore/](https://gestioneaule.unikore.it/agendaweb_unikore/)

---

#### Modalità e orario di ricevimento

Il ricevimento è in presenza (a meno di nuove misure legate ad emergenze sanitarie o altro). Gli orari di ricevimento sono pubblicati sulla pagina personale del docente:

<https://unikore.it/cdl/ingegneria-informatica/persona-e-regolamenti/salvatore-sorce/>

---

<sup>i</sup> PO (professore ordinario), PA (professore associato), RTD (ricercatore a tempo determinato), RU (Ricercatore a tempo indeterminato), DC (Docente a contratto).