



UNIVERSITÀ DEGLI STUDI DI ENNA "KORE"

Facoltà di Ingegneria e Architettura

Anno Accademico 2021/2022

Corso di studi in Ingegneria Informatica, classe di laurea L8

Insegnamento	Linguaggi di Programmazione ad Oggetti
CFU	9
Settore Scientifico Disciplinare	ING-INF/5
Metodologia didattica	Lezioni Frontali
Nr. ore di aula	42
Nr. ore di studio autonomo	167
Nr. ore di laboratorio	16
Mutuazione	Nessuna
Annualità	II
Periodo di svolgimento	Secondo Semestre

Docente	E-mail	Ruolo ⁱ	SSD docente
SINISCALCHI Sabato Marco	marco.siniscalchi@unikore.i	PO	ING-INF/05

Propedeuticità	Fondamenti di Informatica
Sede delle lezioni	Facoltà di Ingegneria e Architettura

Moduli

N.	Nome del modulo	Docente	Durata in ore
----	-----------------	---------	---------------

Orario delle lezioni

Riferirsi al calendario didattico per AA 2021/2022 pubblicato nella pagina web della Facoltà di Ingegneria e Architettura

Obiettivi formativi

Lo scopo del corso 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). Il paradigma orientato agli oggetti e le relative metodologie di programmazione verranno approfondite. Il linguaggio Java sarà approfondito. I class diagram dell'UML (unified modeling language) sono approfonditi.

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: espressività, leggibilità, costo (per imparare un linguaggio), robustezza. Caratteristiche intrinseche dei linguaggi:	Frontale	1h

	modularità, flessibilità, generalità, efficienza, coerenza. Caratteristiche esterne: diffusione, standardizzazione, portabilità, integrabilità.		
10	Introduzione alla programmazione ad oggetti	Frontale	2h
11	Dati numerici in Java (teoria + esempi)	Frontale	1h
12	Definizione di nuove classi in java (teoria + esempi)	Frontale	3h
13	Istruzioni di selezione e ripetizione in Java (teoria + esempi)	Frontale	2h
14	Definizione di classi definite dall'utente in Java (teoria + esempi)	Frontale	4h
15	Esercitazione in laboratorio su argomenti trattati	Laboratorio	8h (23)
16	Gestione delle eccezioni in Java (teoria + esempi)	Frontale	2h
17	Caratteri e Stringhe in Java: Le classi StringBuffer e StringBuilder (teoria ed esempi)	Frontale	2h
18	Array e Collezioni in Java (teoria + esempi)	Frontale	3h
19	Input e Output su file in Java (teoria + esempi)	Frontale	2h
20	Ereditarietà e Polimorfismo in Java (teoria + esempi)	Frontale	5h
21	Programmazione a eventi in Java (GUI) (teoria + esempi)	Frontale	2h
22	Tipi generici e contenitori in Java (teoria + esempi)	Frontale	3h
23	Introduzione all'ambiente di sviluppo e alla compilazione in Java. Esercitazione di base su dati numerici, istruzioni di selezione, istruzioni di ripetizione, e uso delle classi predefinite in Java	Frontale	1h
24	Programmi in Java con classi definite dall'utente	Frontale	1h
25	Programmi per la gestione delle eccezioni, la manipolazione delle stringhe e il meccanismo di gestione dei file in Java	Frontale	1h
26	Sviluppo e implementazione di programmi in Java con uso di collezioni e tipi generici	Frontale	2h
27	Sviluppo e implementazione di programmi in Java con gestione dell'ereditarietà e polimorfismo	Frontale	1h
28	GUI in Java	Frontale	2h
29	Esercitazione in laboratorio su argomenti trattati	Laboratorio	8h

Risultati di apprendimento (descrittori di Dublino)

Alla fine del corso, 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: Lo studente alla fine del corso acquisirà 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 valutarne 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 ed esprimere problematiche inerenti all'oggetto del corso 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 impartite durante il corso.

Testi per lo studio della disciplina

Testi principali: Il Linguaggio Java 4/Ed. K. Arnold - J. Gosling - D. Holmes. Pearson-Addison Wesley, 2006. ISBN: 9788871922768.

Java – Fondamenti di programmazione. C. Thomas Wu McGraw-Hill Italia, 2009. ISBN 88-386-6261-4. ISBN: 9788838665332

Materiale didattico a disposizione degli studenti: Ludici relativi ai principi di programmazione. Esercizi svolti per ogni argomento trattato durante il corso. Esempi di progettazione software in Java. Elenco proposte progettuali per sostenere la prova finale.

Testi di riferimento: Java – Fondamenti di programmazione. C. Thomas Wu McGraw-Hill Italia, 2009. ISBN 88-386-6261-4. ISBN: 9788838665332.

Testi di approfondimento: Linguaggi di programmazione: principi e paradigmi. Maurizio Gabbriellini, Simone Martini. McGraw-Hill Italia, 2006. ISBN 88-386-6261-4.

Modalità di accertamento delle competenze

L'accertamento delle competenze avverrà attraverso una prova orale strutturata nella seguente maniera:

1) una discussione sulla progettazione e messa a punto, 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 assegnato ad ogni gruppo 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 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.

AVVERTENZA: La composizione dei gruppi di lavoro (3 componenti al massimo) per la progettazione e messa a punto del progetto dovrà essere

comunicata via email (marco[.]siniscalchi[et]unikore[.]it) entro un mese dalla data in cui si desidera sostenere l'esame e deve contenere i nomi dei

componenti insieme ai rispettivi indirizzi di posta elettronica e la matricola. La discussione del progetto deve durare 10' per ogni componente del

gruppo. Così un gruppo formato da tre persone dovrà svolgere una presentazione della durata di 30'. Dopo un inquadramento delle problematiche

del progetto bisogna approfondire l'argomento evidenziando gli aspetti peculiari dello stesso. La presentazione deve utilizzare sistemi tipo

powerpoint o equivalenti. All'inizio dell'esame occorre consegnare una breve relazione su carta (4 o 5 fogli) contenente la spiegazione del

problema e della soluzione proposta e un CD contenente la versione finale del progetto in JAVA, il materiale raccolto, la presentazione in ppt e la

relazione in formato elettronico.

L'accertamento delle competenze avverrà attraverso una prova orale interamente basata sul progetto svolto dallo studente (o da un gruppo di studenti nel caso di lavoro in gruppo.) Inoltre si prevede l'accertamento delle conoscenze del linguaggio Java per mezzo di prova al computer.

Date di esame

Le date degli esami saranno pubblicati sulla pagina web del corso di laurea almeno due mesi prima dell'inizio della sessione d'esami.

<https://www.unikore.it/index.php/ingegneria-informatica-esami/calendario-esami>

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, un'ora per CFU erogato, sono pubblicati sulla pagina personale del docente:

<http://www.unikore.it/index.php/ingegneria-informatica-persone/docenti-del-corso/itemlist/category/1589-siniscalchi>

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