



UNIVERSITÀ DEGLI STUDI DI PALERMO

DIPARTIMENTO	Ingegneria
ANNO ACCADEMICO OFFERTA	2022/2023
ANNO ACCADEMICO EROGAZIONE	2023/2024
CORSO DILAUREA	INGEGNERIA CIBERNETICA
INSEGNAMENTO	PROGRAMMAZIONE
TIPO DI ATTIVITA'	A
AMBITO	50283-Matematica, informatica e statistica
CODICE INSEGNAMENTO	05871
SETTORI SCIENTIFICO-DISCIPLINARI	ING-INF/05
DOCENTE RESPONSABILE	LA CASCIA MARCO Professore Ordinario Univ. di PALERMO
ALTRI DOCENTI	
CFU	9
NUMERO DI ORE RISERVATE ALLO STUDIO PERSONALE	144
NUMERO DI ORE RISERVATE ALLA DIDATTICA ASSISTITA	81
PROPEDEUTICITA'	
MUTUAZIONI	
ANNO DI CORSO	2
PERIODO DELLE LEZIONI	1° semestre
MODALITA' DI FREQUENZA	Facoltativa
TIPO DI VALUTAZIONE	Voto in trentesimi
ORARIO DI RICEVIMENTO DEGLI STUDENTI	LA CASCIA MARCO Lunedì 15:00 17:00 Microsoft Teams Codice: wztkv0u

<p>PREREQUISITI</p>	<p>Concetto di algoritmo. Concetti di rappresentazione delle informazioni nei calcolatori. Conoscenza di semplici strutture dati. Conoscenza di base del linguaggio C.</p>
<p>RISULTATI DI APPRENDIMENTO ATTESI</p>	<p>- Conoscenza e capacita' di comprensione Lo studente al termine del corso avra' conoscenza delle problematiche inerenti la programmazione e la progettazione ad oggetti e lo sviluppo e manutenzione di software scritto in Java. In particolare lo studente conoscerà nel dettaglio caratteristiche del linguaggio Java come operatori, funzioni, strutture di controllo, classi e oggetti, ereditarietà, polimorfismo, interfacce, metodi e classi generiche, gestione delle eccezioni, collezioni, sistema di I/O. Per il raggiungimento di questo obiettivo il corso comprende: lezioni frontali; analisi e discussione di frammenti di programmi.</p> <p>- Capacita' di applicare conoscenza e comprensione Lo studente sarà in grado di progettare e sviluppare software complessi utilizzando le caratteristiche avanzate di Java; saprà intervenire su software esistente al fine di correggerne o incrementarne le funzionalità. Per il raggiungimento di questo obiettivo il corso comprende esercitazioni da svolgere in aula informatica comprendenti la scrittura di programmi in Java.</p> <p>- Autonomia di giudizio Lo studente sarà in grado di analizzare classi e librerie di terze parti e valutarne la possibile adozione nello sviluppo di software complessi. Per il raggiungimento di questo obiettivo il corso comprende: analisi e discussioni su casi di studio; una presentazione sommaria della libreria standard e delle principali strutture dati e discussioni su possibili vantaggi e svantaggi derivanti dal loro uso.</p> <p>- Abilita' comunicative Lo studente acquisirà la capacita' di comunicare ed esprimere problematiche inerenti l'oggetto del corso. Sarà in grado di sostenere conversazioni su tematiche di sviluppo software, di evidenziare problemi relativi a progetto e implementazione e di offrire soluzioni. Per il raggiungimento di questo obiettivo il corso comprende esercitazioni in aula informatica in cui gli studenti espongono come risolvono gli esercizi e le eventuali difficoltà incontrate.</p> <p>- Capacita' d'apprendimento Lo studente sarà in grado di affrontare in autonomia qualsiasi problematica relativa allo sviluppo di software con linguaggi ad oggetti. Sarà in grado di approfondire tematiche complesse quali il polimorfismo, la gestione dinamica della memoria, la gestione della concorrenza, la scalabilità del software, etc... Per il raggiungimento di questo obiettivo il corso comprende: esercitazioni da svolgere autonomamente; discussione sulle eventuali difficoltà incontrate.</p>
<p>VALUTAZIONE DELL'APPRENDIMENTO</p>	<p>Il voto finale sarà espresso in trentesimi e varierà da 18/30 a 30/30 con lode. Sono previste due prove: una prova pratica (in aula informatica) ed una discussione orale. Durante la prova pratica sarà chiesto al candidato di scrivere un programma completo in Java a partire dalla descrizione testuale del problema. I candidati che raggiungono la sufficienza (18/30) nella prova pratica sono ammessi a sostenere la discussione orale sugli argomenti del programma. La discussione orale mirerà ad evidenziare in quale misura il candidato ha compreso ed è in grado di padroneggiare i concetti base della programmazione ad oggetti. Il voto finale sarà calcolato come media tra i voti ottenuti nella prova pratica e nella discussione orale. In accordo con i descrittori di Dublino, la formulazione delle prove permette una valutazione dei risultati attesi in relazione al voto finale come segue: - da 18/30 a 20/30: mediocre o sufficiente conoscenza e capacita' di comprensione degli argomenti trattati, parziale capacita' di applicazione delle conoscenze acquisite per la risoluzione dei problemi proposti; parziale autonomia di giudizio, abilita' comunicative e capacita' di apprendere. - da 21/30 a 23/30: sufficiente o discreta conoscenza e capacita' di comprensione degli argomenti trattati, sufficiente capacita' di applicazione delle conoscenze acquisite per la risoluzione dei problemi proposti, sufficiente autonomia di giudizio, abilita' comunicative e capacita' di apprendere. - da 24/30 a 26/30: discreta conoscenza e capacita' di comprensione degli argomenti trattati, discreta capacita' di applicazione delle conoscenze acquisite per la risoluzione dei problemi proposti, sufficiente autonomia di giudizio, abilita' comunicative e capacita' di apprendere. - da 27/30 a 30/30 e lode: buona o eccellente conoscenza e capacita' di comprensione degli argomenti trattati, buona o eccellente capacita' di applicazione delle conoscenze acquisite per la risoluzione dei problemi proposti, buona o eccellente autonomia di giudizio, abilita' comunicative e capacita' di</p>

	<p>apprendere.</p> <p>Requisito minimo per il superamento dell'esame e' la dimostrata conoscenza, nelle due prove, delle nozioni relative a classi e oggetti, ereditarieta, polimorfismo, gestione delle eccezioni.</p>
OBIETTIVI FORMATIVI	<p>Il corso tratta in maniera approfondita la programmazione ad oggetti in Java. Verranno trattati sia gli aspetti relativi alla programmazione ad oggetti in generale che aspetti peculiari del linguaggio Java quali l'organizzazione del codice in package, la documentazione del codice, le funzionalita' per la programmazione concorrente e la libreria standard. Verranno infine trattate le strutture dati di uso piu' frequente inquadrando nel contesto delle libreria Java.</p>
ORGANIZZAZIONE DELLA DIDATTICA	<p>Lezioni frontali, esercitazioni in aula informatica</p>
TESTI CONSIGLIATI	<p>CLAUDIO DE SIO CESARI (2020). Il nuovo Java. Guida completa alla programmazione moderna. Hoepli. ISBN 978-88-203-9930-6</p>

PROGRAMMA

ORE	Lezioni
2	Introduzione a Java
4	Componenti fondamentali di un programma Java: le basi della programmazione object-oriented, i metodi in Java, le variabili, i metodi costruttori, i package.
4	Identificatori, tipi di dati e array: i tipi di dati primitivi e non primitivi, gli array in Java.
4	Operatori e gestione del flusso di esecuzione: gli operatori di base, la gestione del flusso di esecuzione, costrutti di programmazione semplici e avanzati.
6	Incapsulamento e visibilita: i paradigmi della programmazione a oggetti, l'incapsulamento, i modificatori d'accesso, il modificatore static.
6	Ereditarieta' e interfacce: l'ereditarieta, il modificatore final, la classe Object, ereditarieta' e incapsulamento, le interfacce.
6	Polimorfismo: il polimorfismo per metodi, overload e override, il polimorfismo per dati.
4	Eccezioni e asserzioni: il meccanismo per la gestione delle eccezioni in Java, la propagazione dell'eccezione, introduzione alle asserzioni.
6	Collezioni: interfacce Collection, List, Queue e Deque, Map e SortedMap, algoritmi di utilita.
2	Tipi generici: creare propri tipi generici, la deduzione automatica del tipo.
2	Enumerazioni e tipi innestati: le classi interne, le classi anonime, i tipi enumerazione.
4	Libreria: le classi String, Object, System, Runtime e Math, il package java.util, Date-Time API.
4	Input/Output: character e byte stream, la lettura di input da tastiera, la gestione dei files, la serializzazione di oggetti.

ORE	Esercitazioni
3	Ambiente di sviluppo e compilazione. Semplici programmi in Java.
4	Implementazione di semplici classi.
6	Implementazione di gerarchie di classi e polimorfismo.
2	Programmi che usano Eccezioni e implementazione di Eccezioni personalizzate.
3	Uso della libreria di Java
6	Programmi che utilizzano collezioni e il sistema di I/O.
3	Sviluppo di prove di esami di anni precedenti.