Denominazione	PROGRAMMAZIONE OBJECT ORIENTED
Moduli componenti	-
Settore scientifico-disciplinare	IINF-05/A
Anno di corso e semestre di erogazione	2° anno, 1° semestre
Lingua di insegnamento	Italiano
Carico didattico in crediti formativi universitari	6
Numero di ore di attività didattica frontale	48 (pari a 6 CFU di didattica erogativa)
Docente	Giovanni Maria Biancofiore
Risultati di apprendimento specifici	Il corso mira a fornire agli studenti una solida comprensione dei concetti di base della programmazione orientata agli oggetti (OO) e del relativo paradigma attraverso l'uso di linguaggi object-oriented come Python. *Conoscenze e comprensione.* Lo studente acquisirà le conoscenze relative a: modellazione delle informazioni seguendo il paradigma OO; metodologie e tecniche avanzate di programmazione; linguaggi di programmazione orientati agli oggetti (Python). *Capacità di applicare conoscenze e comprensione.* Lo studente svilupperà competenze per essere in grado di: studiare e valutare autonomamente un problema e le sue possibili soluzioni; ideare e formalizzare una soluzione algoritmica sfruttando il paradigma OO; saper scegliere la soluzione algoritmica con il miglior rapporto costi/benefici. *Autonomia di giudizio e pensiero critico:* Al termine del corso lo studente sarà in grado di gestire problematiche relative alla formulazione di algoritmi per la soluzione dei problemi posti e la loro implementazione nel linguaggio di programmazione di riferimento. Saprà inoltre comunicare con chiarezza le metodologie utilizzate e i risultati ottenuti. *Abilità comunicative:* Lo studente sarà in grado di documentare in modo chiaro e strutturato il codice sorgente e i progetti software realizzati. Saprà comunicare efficacemente con colleghi e collaboratori, anche in contesti multidisciplinari, utilizzando un linguaggio tecnico appropriato e diagrammi per rappresentare la struttura e il comportamento del software. *Capacità di apprendimento:* Lo studente saprà aggiornarsi autonomamente sulle evoluzioni del paradigma OO e dei linguaggi di programmazione ad oggetti. Sarà in grado di consultare documentazione tecnica, risorse online, manuali e tutorial per approfondire argomenti avanzati o affrontare nuove tecnologie orientate agli oggetti.

Programma	Il programma del corso è composto dai seguenti contenuti didattici:
	• Teoria della programmazione orientata agli oggetti
	  ○ Principi e caratteristiche OOP
	  ○ Classi, oggetti, attributi e metodi
	  ○ Diagramma delle classi UML

	○ Costruttori e Distruttori ○ Polimorfismo, Overloading vs Overriding, Operator Overloading ○ Incapsulamento, Variabili, locali, non locali e globali ○ Sottoclassi, Ereditarietà e Gerarchie, Composizione, ereditarietà multipla ○ Interfacce, Abstract Base Classes ○ Implementation Inheritance vs Interface Inheritance ○ Generics e Design Patterns Creazionali, Strutturali e Comportamentali • Algoritmi di ricorsione e ordinamento • Gestione dei file testuali e binari • Programmazione in Python e costrutti specifici del linguaggio (Builtins e Container)
Tipologie di attività didattiche previste e relative modalità di svolgimento	L'insegnamento è strutturato in lezioni di didattica frontale, incoraggiando l'interazione e la partecipazione attiva degli studenti, ed esercitazioni, integrate con le lezioni e svolte con l'ausilio di un elaboratore. Saranno mostrati esempi di progettazione ed implementazione di applicativi basati su programmazione orientata agli oggetti. È previsto inoltre l'utilizzo di tecnologie digitali per l'erogazione delle lezioni e delle esercitazioni.
Metodi e criteri di valutazione dell'apprendimento	La valutazione dell'apprendimento (sia per studenti frequentanti che non frequentanti) consiste nello svolgimento di una prova orale contenente quesiti teorici e pratici relativi ai contenuti del corso proposti attraverso domande frontali. Sarà richiesta inoltre la definizione e codifica di un algoritmo in linguaggio Python su base progettuale. La valutazione del quesito progettuale terrà conto della comprensione delle specifiche di progetto e della qualità implementativa della soluzione proposta, oltre che della documentazione realizzata e della esposizione orale in sede d'esame. La prova progettuale concorrerà alla composizione del voto finale, nella misura del 40%. Il restante 60% della valutazione si baserà sul colloquio orale. La valutazione del colloquio orale è espressa in diciottesimi e terrà conto della proprietà di linguaggio, della capacità argomentativa, di analisi critica e di ragionamento.
Criteri di misurazione dell'apprendimento e di attribuzione del voto finale	La valutazione dell'apprendimento prevede l'attribuzione di un voto finale espresso in trentesimi. Il voto finale sarà determinato attraverso l'esame scritto sopra dettagliato. La lode sarà valutata per i soli studenti che abbiano raggiunto la valutazione complessiva di 30/30 analizzando la capacità di proporre soluzioni corrette ed efficienti nella risoluzione di problemi.
Propedeuticità	Fondamenti di Informatica N.B.: Come indicato dal regolamento didattico del corso di studio, è richiesto il **superamento** degli esami di "Fondamenti di Informatica" per poter sostenere l'esame di "Programmazione Object Oriented".
Materiale didattico utilizzato e materiale didattico consigliato	– Report, documenti e slides fornite dal docente – Hands-On Data Structures and Algorithms with Python - Second Edition, Basant Agarwal, Benjamin Baka – Introduzione a PYTHON, Tony Gaddis, Edizioni Pearson – Programmazione in Python, Kenneth A. Lambert, Apogeo Education