



**Università degli Studi di Enna "Kore"**  
**Facoltà di Ingegneria e Architettura**  
**Anno Accademico 2019 - 2020**

A.A.	Settore Scientifico Disciplinare		CFU	Insegnamento	Ore di aula		Mutuazione	
2019/20	ING-INF/05		6	<b>Algoritmi e Strutture Dati</b>	48		No	
Classe	Corso di studi			Tipologia di insegnamento	Anno di corso e Periodo		Sede delle lezioni	
L8	Ingegneria Informatica			Caratterizzante	II Anno Primo Semestre		Plesso di Ingegneria	
N° Modulo	Nome Modulo	Tipologia lezioni	Ore	Docente	SSD	Ruolo	Interno	Affidamento
No		Lezioni frontali	48	Giovanni Pau <a href="mailto:giovanni.pau@unikore.it">giovanni.pau@unikore.it</a>	ING-INF/05	RTD	Si	Istituzionale

### Prerequisiti

Per una corretta fruizione del corso, è auspicabile che lo studente abbia già acquisito conoscenze, capacità ed abilità teoriche e/o applicate relative al corso di Fondamenti di Informatica e Laboratorio.

### Propedeuticità

Nessuna.

### Obiettivi formativi

L'obiettivo del corso è fornire agli studenti gli strumenti teorici e pratici, attraverso il linguaggio Python, per affrontare la progettazione di soluzioni algoritmiche corrette ed efficienti per un'ampia gamma di problemi computazionali.



*Università degli Studi di Enna "Kore"*  
*Facoltà di Ingegneria e Architettura*

**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:** lo studente alla fine del corso acquisirà una buona conoscenza di algoritmi e delle più importanti strutture dati utilizzate nella programmazione avanzata. Inoltre, sarà in grado di analizzare e comprendere il codice sorgente in linguaggio Python di diversi algoritmi utilizzati per lo sviluppo del software.

**Conoscenza e capacità di comprensione applicate:** lo studente sarà in grado di valutare le caratteristiche, i vantaggi e le limitazioni di diversi algoritmi e strutture dati. Inoltre, attraverso il linguaggio Python, sarà in grado non solo di progettare, analizzare e valutare le soluzioni software a problemi di media complessità ma anche di sviluppare nuove soluzioni software, valutandone la qualità in termini di semplicità, efficacia ed efficienza.

**Autonomia di giudizio:** lo studente sarà in grado di effettuare l'analisi sulle problematiche relative alla progettazione software. Inoltre, a partire da precise specifiche, sarà in grado di implementare opportune soluzioni software mediante il linguaggio Python, valutandone la qualità in termini di semplicità, leggibilità, efficienza e possibilità di riutilizzo.

**Abilità comunicative:** lo studente acquisirà la capacità di comunicare ed esprimere problematiche inerenti l'oggetto del corso. Sarà in grado di sostenere conversazioni su tematiche relative alle implementazioni software di algoritmi e strutture dati efficienti. Inoltre, sarà in grado di utilizzare un linguaggio semplice e chiaro per la descrizione dei processi di analisi e di sintesi di soluzioni software a problemi di media complessità.

**Capacità di apprendere:** lo studente svilupperà la capacità di apprendere i processi di analisi e di sintesi relativi alla codifica di algoritmi di media complessità ed alla relativa implementazione mediante il linguaggio Python.



**Università degli Studi di Enna "Kore"**  
**Facoltà di Ingegneria e Architettura**

**Contenuti e struttura del corso**

N.	ARGOMENTO	TIPOLOGIA	DURATA
1	<ul style="list-style-type: none"><li>• <i>Caratteristiche di algoritmi e strutture dati</i></li><li>• <i>Python: la comunità di Python, operazioni preliminari, installazione, IDLE e shell, interprete</i></li></ul>	Frontale	3h
2	<ul style="list-style-type: none"><li>• <i>Python: commenti, valutazione espressioni, operazioni aritmetiche, variabili e operatori, funzioni predefinite, tipi di dato, stile di programmazione</i></li><li>• <i>Esempi ed esercizi</i></li></ul>	Frontale	3h
3	<ul style="list-style-type: none"><li>• <i>Python: sequenze e flow chart, controllo e selezione, blocchi di codice e indentazione, operatori logici</i></li><li>• <i>Esempi ed esercizi</i></li></ul>	Frontale	5h
4	<ul style="list-style-type: none"><li>• <i>Python: moduli, funzioni e numeri casuali, gestione variabili locali e globali, ambiente grafico turtle e relativi metodi, cicli</i></li><li>• <i>Esempi ed esercizi</i></li></ul>	Frontale	5h
5	<ul style="list-style-type: none"><li>• <i>Python: stringhe, liste, tuple, insiemi e dizionari</i></li><li>• <i>Esempi ed esercizi.</i></li></ul>	Frontale	4h
6	<ul style="list-style-type: none"><li>• <i>Python: errori, test e debugging, file, dati e statistica</i></li><li>• <i>Esempi ed esercizi</i></li></ul>	Frontale	3h
7	<ul style="list-style-type: none"><li>• <i>Analisi di algoritmi, strutture dati fondamentali</i></li></ul>	Frontale	5h
8	<ul style="list-style-type: none"><li>• <i>Pile, code e code doppie, liste concatenate, alberi, code prioritarie, mappe, tabelle di hash, alberi di ricerca</i></li></ul>	Frontale	14h
9	<ul style="list-style-type: none"><li>• <i>Ordinamento e selezione, algoritmi per grafi</i></li></ul>	Frontale	6h

**Attività esercitative / Lavoro di gruppo:**

Presentazione di esercizi e relative soluzioni e casi di studio su argomenti trattati durante il corso.



*Università degli Studi di Enna "Kore"*  
*Facoltà di Ingegneria e Architettura*

## Testi adottati

### Testi principali:

Maurizio Boscaini, *"Imparare a programmare con Python - il manuale per programmatori dai 13 anni in su,"* Apogeo  
<http://www.apogeoonline.com/libri/9788850333981/scheda>

Michael T. Goodrich - Roberto Tamassia - Michael H. Goldwasser, *"Data structures and Algorithms in Python,"* Wiley  
[https://www.amazon.it/Structures-Algorithms-Python-Michael-Goodrich/dp/812656217X/ref=tmm\\_pap\\_swatch\\_0?encoding=UTF8&qid=&sr=](https://www.amazon.it/Structures-Algorithms-Python-Michael-Goodrich/dp/812656217X/ref=tmm_pap_swatch_0?encoding=UTF8&qid=&sr=)

### Materiale didattico a disposizione degli studenti:

Le slide delle lezioni sono scaricabili online dalla pagina del docente.

### Testi di approfondimento:

Michael T. Goodrich - Roberto Tamassia - Michael H. Goldwasser, *"Algoritmi e strutture dati in Java,"* Apogeo  
<http://www.apogeoeducation.com/9788891613394-algoritmi-e-strutture-dati-in-java.html>

Alan A. Bertossi, Alberto Montresor, *"Algoritmi e strutture di dati,"* CittàStudi  
<https://www.libreriauniversitaria.it/algoritmi-strutture-dati-bertossi-alan/libro/9788825173956>

Cay Horstmann - Rance D. Nicaise, *"Concetti di informatica e fondamenti di Python,"* Apogeo  
<http://www.apogeoeducation.com/informatica/concetti-di-informatica-e-fondamenti-di-python.html>



**Università degli Studi di Enna "Kore"**  
**Facoltà di Ingegneria e Architettura**

**Modalità di accertamento delle competenze**

L'accertamento delle competenze avverrà attraverso una prova orale interamente basata sulla proposta progettuale dallo studente (o da un gruppo di studenti nel caso di lavoro in gruppo). L'elaborato deve rispettare le direttive, relative all'utilizzo delle strutture dati, che saranno pubblicate dal docente nella sua pagina web. La proposta progettuale è volta a dimostrare l'acquisizione degli argomenti erogati durante le lezioni frontali. La durata del colloquio dipenderà dalla profondità ed ampiezza della proposta progettuale ed indicativamente durerà tra 20 ed i 60 minuti. Ove fosse necessario, gli esaminandi saranno ripartiti in più giornate, secondo un calendario determinato nel giorno dell'appello ovvero, se possibile, anticipatamente sulla base delle prenotazioni pervenute. La calendarizzazione sarà, in tal caso, opportunamente pubblicizzata. La valutazione dell'apprendimento sarà focalizzata sulla valutazione dei risultati attesi, in accordo con i descrittori di Dublino.

La prova di esame si intende superata con una votazione minima di 18/30 quando lo studente dimostra:

- minima conoscenza e comprensione degli argomenti trattati;
- limitata capacità nell'applicazione delle conoscenze acquisite;
- sufficiente capacità espositiva.

La votazione di 30/30, eventualmente con lode, è assegnata quando lo studente dimostra:

- ottima conoscenza e comprensione degli argomenti trattati;
- ottima capacità nell'applicazione delle conoscenze acquisite;
- eccellente capacità espositiva.

La prova di esame si intende non superata se lo studente mostra un livello insufficiente di conoscenza degli argomenti trattati e non dimostra una sufficiente capacità nell'applicazione delle conoscenze acquisite.



**Università degli Studi di Enna "Kore"**  
**Facoltà di Ingegneria e Architettura**

**Orari di lezione e date di esame**

Gli orari di lezione saranno pubblicati sulla pagina web del corso di laurea almeno due mesi prima dell'inizio delle lezioni:

<http://www.unikore.it/index.php/ingegneria-informatica-attivita-didattiche/calendario-lezioni>

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

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

**Modalità e orari di ricevimento**

Gli orari di ricevimento sono pubblicati nella cartella "Curriculum e ricevimento" della pagina personale del docente:

<http://www.unikore.it/index.php/ingegneria-informatica-persone/docenti-del-corso/itemlist/category/2395-prof-pau-giovanni>