



*Università degli Studi di Enna "Kore"*  
*Facoltà di Ingegneria ed Architettura*  
*Anno Accademico 2017 - 2018*

A.A.	Settore Scientifico Disciplinare		CFU	Insegnamento	Ore di aula		Mutuazione	
2017/18	ING-INF/05		6	Algoritmi e Strutture Dati (a scelta)	48		No	
Classe	Corso di studi		Tipologia di insegnamento		Anno di corso e Periodo		Sede delle lezioni	
L8	Ingegneria Informatica e delle Telecomunicazioni		Caratterizzante		2° Anno Secondo Semestre		Facoltà di Ingegneria e Architettura	
N° Modulo	Nome Modulo	Tipologia lezioni	Ore	Docente	SSD	Ruolo	Interno	Affidamento
1		Lezioni frontali	38	Vincenzo Conti <a href="mailto:vincenzo.conti@unikore.it">vincenzo.conti@unikore.it</a> 0935 536445	ING-INF/05	RTD	Si	Istituzionale
		Esercitazioni	10					

### Prerequisiti

Per una corretta fruizione del corso sarebbe auspicabile che lo studente abbia già acquisito conoscenze, capacità ed abilità teoriche e/o applicate sia per quanto riguarda la programmazione procedurale sia per quanto riguarda l'analisi matematica

### Propedeuticità

Nessuna.

### Obiettivi formativi

Studio delle caratteristiche di base, efficienza in termini di tempo di elaborazione e memoria utilizzata, dei principali algoritmi e delle più importanti strutture dati della programmazione avanzata. Introduzione all'ambiente di sviluppo e prototipazione MatLab.



**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 (knowledge and understanding):** Lo studente alla fine del corso acquisirà una buona conoscenza dei principali algoritmi e delle più importanti strutture dati utilizzate nella programmazione avanzata. Sarà in grado di analizzare e comprendere il codice sorgente dei principali algoritmi utilizzati per lo sviluppo del software. La capacità di comprensione dello studente verrà valutata, dopo l'esposizione dei principali concetti, durante le lezioni frontali con un dialogo diretto con gli studenti.

**Capacità di applicare conoscenza e comprensione (applying knowledge and understanding):** Lo studente sarà in grado di valutare le caratteristiche, i vantaggi e le limitazioni dei principali algoritmi e strutture dati. Sarà in grado di progettare, analizzare e valutare le soluzioni software a problemi di media complessità. Sarà anche in grado di sviluppare nuove soluzioni software, valutandone la qualità in termini di semplicità, efficacia ed efficienza. Tale capacità verrà valutata principalmente durante le ore di esercitazione.

**Autonomia di giudizio (making judgements):** Lo studente sarà in grado sia di effettuare l'analisi di un problema che di progettare, a partire da precise specifiche, una opportuna soluzione software. Sarà in grado di valutarne la qualità di una soluzione software in termini di semplicità, leggibilità, efficienza e possibilità di riutilizzo. L'autonomia di giudizio verrà valutata esaminando le soluzioni proposte dagli studenti a problemi software di media complessità. Lo studente verrà incoraggiato inizialmente a trovare e valutare autonomamente soluzioni ai problemi posti, al fine di potere comprendere la qualità e l'utilità delle soluzioni proposte successivamente dal docente.

**Abilità comunicative (communication skills):** Lo studente acquisirà la capacità di comunicare ed esprimere problematiche inerenti all'oggetto del corso. Sarà in grado di sostenere conversazioni su tematiche relative alla implementazioni software di algoritmi e strutture dati efficienti. 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. Il carattere interattivo delle lezioni dovrà permettere la valutazione e il miglioramento delle abilità comunicative dello studente.

**Capacità d'apprendimento (learning skills):** Lo studente dovrà sviluppare la capacità di apprendere i processi di analisi e di sintesi relativi alla codifica di algoritmi di media complessità e alla relativa implementazione di librerie e strumenti software. Il grado di apprendimento sarà valutato non in base alla capacità di memorizzare concetti specifici ma in base alla capacità di ricostruire *ex novo* partendo dal minor numero possibile di idee



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

generali di base soluzioni software ottimali.

## Contenuti e struttura del corso

### Lezioni frontali:

N.	ARGOMENTO	TIPOLOGIA	DURATA
1	Introduzione informale agli algoritmi	Frontale	2h
2	Modelli di calcolo e metodologie di analisi	Frontale	2h
3	Strutture dati elementari	Frontale	4h
4	Algoritmi di ordinamento	Frontale	4h
5	Algoritmi di ricerca	Frontale	4h
6	Selezione e statistiche di ordine	Frontale	2h
7	Alberi di ricerca	Frontale	4h
8	Tabelle hash	Frontale	2h
9	Tecniche algoritmiche	Frontale	4h
10	Grafi e visite di grafi	Frontale	4h
11	Minimo albero ricoprente	Frontale	2h
12	Cammini Minimi	Frontale	4h
13	Esercitazioni (relative a tutti gli argomenti del corso)	Esercitazione	10h

### Attività esercitative / Lavoro di gruppo:

Sviluppo di esercizi relative agli argomenti trattati durante il corso e introduzione alla programmazione e all'ambiente di sviluppo MatLab.

### Testi adottati

#### Testi principali:

C. Demetrescu, I. Finocchi, G.F. Italiano, "Algoritmi e Strutture Dati", Seconda edizione, McGraw-Hill

#### Materiale didattico a disposizione degli studenti:

Slide del corso



## ***Università degli Studi di Enna "Kore"***

### ***Facoltà di Ingegneria e Architettura***

Esercizi svolti per ogni argomento trattato durante il corso

### **Modalità di accertamento delle competenze**

L'obiettivo della prova d'esame consiste nel verificare il livello di raggiungimento delle conoscenze, competenze e abilità in accordo con i descrittori di Dublino. Il voto sarà dato in trentesimi e varierà da 18/30 a 30/30 con lode. L'accertamento delle competenze si basa su un esame espletato in un'unica giornata tramite una prova orale basata sull'esposizione della messa a punto di un progetto basato su particolari strutture dati e algoritmi al fine di risolvere in modo efficiente un dato problema. Il progetto è assegnato prima della fine del corso, di norma una settimana prima, e può essere presentato in uno degli esami già fissati nel calendario didattico. Il progetto può essere svolto o singolarmente o in gruppo.

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

- Ottimo (30-30 e lode): Ottima conoscenza e comprensione degli argomenti trattati e utilizzati al fine di risolvere il problema di ottimizzazione assegnato. Eccellenti capacità espositive.
- Molto buono (26-29): Buona conoscenza e comprensione degli argomenti trattati e utilizzati al fine di risolvere il problema di ottimizzazione assegnato. Ottime capacità espositive.
- Buono (24-25): Buona conoscenza e comprensione degli argomenti trattati e utilizzati al fine di risolvere il problema di ottimizzazione assegnato. Buone capacità espositive.
- Discreto (21-23): Discreta conoscenza e comprensione degli argomenti trattati e limitata capacità di applicarli al fine di risolvere il problema di ottimizzazione assegnato.
- Sufficiente (18-20): Conoscenza minima degli argomenti trattati e limitata capacità di applicare le conoscenze acquisite al fine di risolvere il problema di ottimizzazione assegnato.
- Insufficiente: Manca di una conoscenza accettabile degli argomenti trattati e non dimostra una sufficiente capacità di applicare le conoscenze acquisite al fine di risolvere il problema di ottimizzazione assegnato.

### **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#>



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

### **Modalità e orari di ricevimento**

Gli orari di ricevimento saranno pubblicati sulla pagina personale del docente:

<http://www.unikore.it/index.php/ingegneria-informatica-persone/docenti-del-corso/itemlist/category/1511-conti>

### **Note**

Nessuna.

