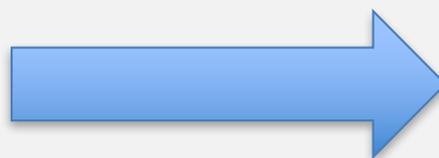


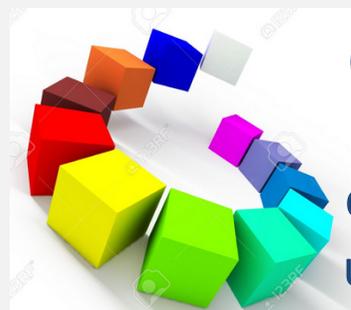
Le **organizzazioni consultate** sono interpellate in merito a funzioni e competenze professionali



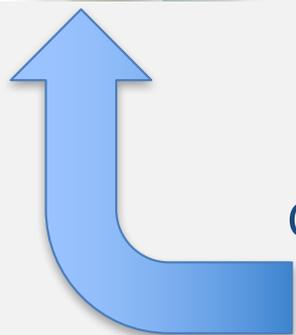
Identificazione di funzioni e competenze dei **profili professionali**



Corrispondenza competenze dei profili professionali definiti e conoscenze e capacità delle attività formative previste ...



Ogni area di apprendimento riporta gli **Insegnamenti**, a ciascuno dei quali è collegata una **Scheda di Insegnamento**



**Are di Apprendimento**  
Per ciascuna, declinate:  
Conoscenza e comprensione;  
Capacità di applicare  
conoscenza e comprensione



... ottenuta in due step (quadri A4.b, e Schede di Insegnamento)

**Così capisco cosa apprendo, e come lo apprendo (e in che modo è svolta la verifica dell'apprendimento, dalla Scheda di Insegnamento)!**



# La Scheda Insegnamento

Tra lo studente e l'Università e il Corso di Studi viene stipulato un patto, anche se non viene formalizzato.

E' fondamentale che lo studente venga informato in modo chiaro su:

- Corso di Studio,
- obiettivi formativi,
- sbocchi professionali.

ma anche su:

- gli obiettivi formativi dei singoli insegnamenti,
- le propedeuticità (non solo quelle amministrative ma, soprattutto, quelle culturali),
- i contenuti dei corsi,
- le modalità didattiche,
- i metodi di verifica dell'apprendimento,
- i testi consigliati.

Queste ultime informazioni sono riportate nella **Scheda Insegnamento**

La scheda insegnamento deve essere:

- facilmente reperibile
- completa
- aggiornata
- chiara

Il Coordinatore del CdS deve:

- verificare che sia stata compilata secondo le linee guida

La **Commissione Paritetica Docenti Studenti** deve:

- verificare (a campione o sulla totalità delle schede)
  - che siano presenti
  - che, sulla base delle informazioni ricevute dagli studenti, le informazioni siano veritiere

La scheda insegnamento ha anche un altro scopo:

il Coordinatore del CdS, attraverso le schede insegnamento, verifica che:

- ci sia corrispondenza tra quanto indicato tra gli obiettivi formativi del corso e quanto viene effettivamente insegnato
- non ci siano argomenti che non sono trattati in nessun insegnamento
- non ci siano duplicati o sovrapposizioni
- le propedeuticità culturali siano effettivamente rispettate



# Esempi di Schede Insegnamento carenti

**CFU: 6**

**SSD: FIS/01**

**Ore di lezione: 38**

**Ore di esercitazione: 14**

**Anno di corso: I**

**Obiettivi formativi:** Lo studente acquisirà i concetti fondamentali dell'Elettromagnetismo, privilegiando gli aspetti fenomenologici e metodologici. Acquisirà inoltre una abilità operativa consapevole nella risoluzione di semplici esercizi numerici.

**Contenuti:** Interazione elettrica. Il principio di conservazione della carica elettrica. Legge di Coulomb. Principio di sovrapposizione. Campo elettrico. Potenziale elettrostatico. Potenziale di dipolo. Forza risultante e momento risultante su un dipolo posto in un campo esterno. Flusso di un campo vettoriale. Legge di Gauss. Il campo elettrico in presenza di conduttori. Condensatori. Densità di energia del campo elettrico. Cenni sull'elettrostatica nei dielettrici. Correnti continue. Legge di Ohm. Legge di Joule. Forza elettromotrice di un generatore. Leggi di Kirchhoff. Circuito RC. Interazione magnetica. Forza di Lorentz. Forza su un conduttore percorso da corrente. Momento meccanico su una spira. Moto di una carica in un campo magnetico uniforme. Il campo magnetico generato da correnti stazionarie. Il campo di una spira a grande distanza. Il momento magnetico di una spira. La legge di Gauss per il magnetismo. Il teorema della circuitazione di Ampere. Cenni sulla magnetostatica nei mezzi materiali. Legge di Faraday. Coefficienti di Auto e Mutua induzione. Circuito RL. Densità di energia del campo magnetico. Corrente di spostamento. Equazioni di Maxwell. Cenni sulle onde elettromagnetiche.



# Esempi di Schede Insegnamento carenti

**Codice:** 117

**Semestre:** secondo

**Propedeuticità:** Fisica Generale I

**Metodo didattico:** lezioni, esercitazioni e prove in itinere

**Materiale didattico:** Libro di testo

**Modalità d'esame:** Scritto ed orale



# Esempi di Schede Insegnamento carenti

Nessun partizionamento

**Corso di studio** Ingegneria Elettronica (D.M.270/04)  
**Ordinamento** INGEGNERIA ELETTRONICA (D.M.270/04)  
**Percorso comune**

**Docenti:**

**Numero ore:** 81

**Periodo:** Secondo Ciclo Semestrale

**Crediti:** 9

**Settori:** ING-INF/01

## **Obiettivi formativi**

Il corso si prefigge l'obiettivo di fornire una descrizione del funzionamento dei principali circuiti elettronici che si trovano al cuore dei sistemi elettronici sia in tecnologia CMOS che in tecnologia bipolare.



# Esempi di Schede Insegnamento carenti

## **Prerequisiti**

Nessuno. Elettronica I fortemente consigliato.

## **Contenuti del corso**

Operational Amplifiers, Differential Amplifiers, Frequency Response, Feedback, Analog Filters, Digital Circuits

## **Metodi didattici**

## **Verifica dell'apprendimento**

## **Testi di riferimento**

"Fundamentals of Microelectronics", Behzad Razavi, John Wiley & Sons Ltd. 2008, ISBN 978-0-471-47846-1

R.C.Jaeger, T.N.Blalock, Microelettronica, Ed. McGraw-Hill, ISBN: 978-88-386-6504-2

## **Risultati di apprendimento attesi**

*Stampa del 18/05/2016*



# Esempi di Schede Insegnamento carenti

## Informazioni generali

ORDINAMENTO	Lauree ai sensi ex D.M. 270/04
CORSO DI LAUREA	<b>Laurea in Ingegneria Elettronica</b>
CDS	<b>Ingegneria Elettronica</b>
ANNO DI CORSO	2
CODICE IDENTIFICATIVO	
SSD	ING-INF/01
CFU	9



# Esempi di Schede Insegnamento carenti

ORE LEZIONI IN AULA	49
ORE ALTRE ATTIVITÀ	30
ORE TOTALI	79
PERIODO	2° Sem
PROGRAMMA SINTETICO	
AVVISO	

## Docenti

DOCENTE	SCAGLIONE	PROGRAMMA DETTAGLIATO	INFORMAZIONI
	A - Z	-	-

Il PQA di UniFE ha iniziato nel 2015 una forte campagna di sensibilizzazione

- nei confronti dei docenti affinché compilino correttamente le schede insegnamento
- nei confronti dei coordinatori di CdS affinché verifichino le schede insegnamento dei loro corsi.

Inoltre sono stati riorganizzati tutti i siti dei CdS in modo che per ciascun insegnamento sia presente un “minisito” che contenga almeno tre informazioni fondamentali:

- scheda insegnamento
- informazioni utili
- materiale didattico

### Fondamenti di informatica e laboratorio

- ▶ Scheda insegnamento Fondamenti di Informatica fino all' a.a. 2016/17
- ▶ Moduli e materiale didattico
- ▶ Orario delle lezioni
- ▶ Esame
- ▶ Informazioni utili

## FONDAMENTI DI INFORMATICA E LABORATORIO

Anno accademico e docente 2017/2018 - EVELINA LAMMA

Non hai trovato la Scheda dell'insegnamento riferita all'anno accademico di tuo interesse? [Ecco come fare >>](#)

### Obiettivi formativi

Il corso è il primo insegnamento di programmazione software. L'obiettivo principale del corso consiste nel fornire agli studenti le basi per individuare l'algoritmo risolutivo di un problema, confrontare l'efficienza di due o più algoritmi e realizzare algoritmi nel linguaggio imperativo C e/o nel linguaggio a oggetti Java, utilizzando semplici strutture di dato.

Il corso introduce gli elementi di base di un elaboratore per il supporto alla programmazione, il C come primo linguaggio di programmazione software, e la sua macchina virtuale. Alla conoscenza del linguaggio C si affianca la presentazione di tecniche algoritmiche – anche ricorsive - per risolvere problemi tipici dell'informatica, discutendone la complessità, e l'uso di semplici strutture di dati, anche dinamiche, per la risoluzione software di problemi di elaborazione. Partendo dalla nozione di progetto software su più file sorgenti, introducendo la nozione di programmazione a moduli e object-oriented, si presenta poi il linguaggio a oggetti Java e le sue caratteristiche rispetto al linguaggio C. Infine, le principali strutture dati dinamiche realizzate in C sono presentate come classi e interfacce della Java Collections Framework.

Le principali conoscenze acquisite saranno:

- nozione di algoritmo, e calcolo della complessità di un algoritmo
- architettura della macchina di Von Neumann, e macchina astratta a supporto dell'esecuzione di un programma in linguaggio C
- il linguaggio di programmazione C
- la realizzazione in C delle principali strutture di dato, anche dinamiche
- la programmazione su più file
- la programmazione a oggetti e il linguaggio Java
- la Java Collections Framework come implementazione in Java di strutture di dato.

Le principali abilità (ossia la capacità di applicare le conoscenze acquisite) saranno:

- capacità di risolvere semplici problemi con tecniche algoritmiche e realizzarle mediante il linguaggio di programmazione C e/o Java
- essere in grado di individuare l'evoluzione a tempo di esecuzione di un programma in linguaggio C
- essere in grado di valutare la complessità di semplici algoritmi, anche su strutture dati collegate (liste, alberi).

### English course description

**Anno accademico**

2017/2018

**Docente**

EVELINA LAMMA

**Crediti formativi**

15

**Periodo didattico**

Annuale

**SSD**

ING-INF/05



# Schede Insegnamento e Diploma Supplement

Il Diploma Supplement è un documento che contiene tutte le informazioni dettagliate sul CdS e sugli esami superati da parte di uno studente.

E' un documento molto importante nei casi di trasferimento in Italia o in Europa

I contenuti della scheda insegnamento sono sincronizzati ai contenuti del Diploma Supplement, in quanto il docente aggiorna i dati sul Diploma Supplement e questi sono automaticamente travasati sulla scheda insegnamento, sul portale web



# I contenuti: obiettivi formativi

## FONDAMENTI DI INFORMATICA E LABORATORIO

Anno accademico e docente 2017/2018 - EVELINA LAMMA

Non hai trovato la Scheda dell'insegnamento riferita all'anno accademico di tuo interesse? Ecco come fare >>

### Obiettivi formativi

Il corso è il primo insegnamento di programmazione software. L'obiettivo principale del corso consiste nel fornire agli studenti le basi per individuare l'algoritmo risolutivo di un problema, confrontare l'efficienza di due o più algoritmi e realizzare algoritmi nel linguaggio imperativo C e/o nel linguaggio a oggetti Java, utilizzando semplici strutture di dato.

Il corso introduce gli elementi di base di un elaboratore per il supporto alla programmazione, il C come primo linguaggio di programmazione software, e la sua macchina virtuale. Alla conoscenza del linguaggio C si affianca la presentazione di tecniche algoritmiche – anche ricorsive - per risolvere problemi tipici dell'informatica, discutendone la complessità, e l'uso di semplici strutture di dati, anche dinamiche, per la risoluzione software di problemi di elaborazione. Partendo dalla nozione di progetto software su più file sorgenti, introducendo la nozione di programmazione a moduli e object-oriented, si presenta poi il linguaggio a oggetti Java e le sue caratteristiche rispetto al linguaggio C. Infine, le principali strutture dati dinamiche realizzate in C sono presentate come classi e interfacce della Java Collections Framework.

#### Le principali conoscenze acquisite saranno:

- nozione di algoritmo, e calcolo della complessità di un algoritmo
- architettura della macchina di Von Neumann, e macchina astratta a supporto dell'esecuzione di un programma in linguaggio C
- il linguaggio di programmazione C
- la realizzazione in C delle principali strutture di dato, anche dinamiche
- la programmazione su più file
- la programmazione a oggetti e il linguaggio Java
- la Java Collections Framework come implementazione in Java di strutture di dato.

#### Le principali abilità (ossia la capacità di applicare le conoscenze acquisite) saranno:

- capacità di risolvere semplici problemi con tecniche algoritmiche e realizzarle mediante il linguaggio di programmazione C e/o Java
- essere in grado di individuare l'evoluzione a tempo di esecuzione di un programma in linguaggio C
- essere in grado di valutare la complessità di semplici algoritmi, anche su strutture dati collegate (liste, alberi).

#### English course description

##### Anno accademico

2017/2018

##### Docente

EVELINA LAMMA

##### Crediti formativi

15

##### Periodo didattico

Annuale

##### SSD

ING-INF/05



# I contenuti: prerequisiti

## **Prerequisiti**

Comprensione del testo

Conoscenze minime di matematica come da scuola media superiore

Capacità di ragionamento logico

Operazioni logiche (and, or, not)



# I contenuti: contenuti del corso

## Contenuti del corso

Il corso include ore di didattica nella forma di lezioni ed esercitazioni. 6 CFU sono occupati dal Modulo A dell'insegnamento, e 9 CFU dal Modulo B.

Nel Modulo A, due terzi delle ore di lezione sono erogate in aula e un terzo come ore di esercitazioni guidate in laboratorio. Gli argomenti affrontati sono nel Modulo A sono:

Nozione di algoritmo

La Macchina di Von Neumann

La programmazione strutturata

Il linguaggio C, le sue strutture di controllo, tipi primitivi, vettori, struct, puntatori, funzioni e passaggio dei parametri, iterazione e ricorsione

Il modello "run-time" del C, argomenti della linea di comando, la programmazione su più file.

Nel Modulo B due terzi delle ore sono dedicate a lezioni (sia in aula sia in Laboratorio di informatica) (indicate di seguito) durante le quali gli studenti verificano direttamente, programmando, ciò che il docente introduce nel corso della lezione (con un approccio learning-by-doing). Il restante terzo è dedicato ad esercitazioni pratiche di consolidamento degli argomenti legati alla programmazione e alle strutture dati presentanti a lezione.

Gli argomenti affrontati nel Modulo B sono:

Complessità degli algoritmi e calcolo della complessità temporale

Algoritmi e strutture di dato anche dinamiche in C

Programmazione in C di algoritmi con strutture dati di tipo lista e albero (in Laboratorio)

La programmazione per componenti e ad oggetti.

Progettare per astrazioni. Incapsulamento e protezione. Classi ed ereditarietà.

Il linguaggio Java (in Laboratorio):

Classe e istanza, costruzione e distruzione di oggetti, oggetti semplici e oggetti composti.

Package e Package di I/O.

Ereditarietà, Classi Astratte, Interfacce.

Eccezioni.

Le librerie grafiche AWT e Swing.

Realizzazione di Applet

Java Collections Framework.



# I contenuti: metodi didattici

## **Metodi didattici**

Il corso comprende lezioni frontali in aula per alcuni argomenti, e lezioni ed esercitazioni pratiche in Laboratorio di Informatica, per argomenti legati alla programmazione in C e in Java.

Nelle ore di lezione ed esercitazione in Laboratorio inserite nel corso, o previste come tutorato didattico, gli studenti possono usufruire di tutor didattici in numero congruo per essere seguiti nello svolgimento degli esercizi proposti. Gli studenti hanno anche libero accesso al Laboratorio per ulteriori esercitazioni individuali.

## Modalità di verifica dell'apprendimento

L'obiettivo della prova d'esame è verificare il livello di raggiungimento degli obiettivi formativi precedentemente indicati.

L'esame complessivo è costituito da tre prove che hanno luogo nello stesso giorno:

- Una prova di laboratorio, in cui si chiede allo studente di scrivere un programma in linguaggio C, volta a valutare la capacità di risolvere semplici problemi da parte dello studente identificando un algoritmo risolutivo e programmandolo in C, applicando le conoscenze acquisite durante il corso e durante le lezioni di laboratorio.
- Una seconda prova di laboratorio verte sul linguaggio Java ed è volta a valutare la conoscenza dello studente sulla programmazione orientata agli oggetti e Java, acquisita durante il corso e durante le lezioni di laboratorio.
- Una terza prova, di tipo scritto (ma che può essere in parte svolta sempre al computer, con risposte a domande aperte), ha lo scopo di verificare la conoscenza dello studente sul funzionamento della macchina astratta alla base dell'esecuzione di un programma in C, la capacità di utilizzare programmazione ricorsiva e di saper calcolare la complessità di un algoritmo/programma. La prova verifica l'acquisizione delle conoscenze teoriche sugli argomenti del corso. Contiene esercizi di analisi di un programma, programmazione ricorsiva, analisi della complessità e domande a risposta aperta su architettura degli elaboratori e programmazione ad oggetti.

Le prove di laboratorio sono volte a verificare le capacità di sintesi di algoritmi e di programmazione. La prova scritta (parzialmente svolta sempre al computer) ha esercizi volti a verificare le capacità analitiche e le conoscenze su argomenti di base.

Il voto finale è dato dalla somma dei punteggi delle tre prove.

Per superare l'esame è necessario acquisire un punteggio minimo di 18 su 31.

Qualora una delle tre prove risulti insufficiente o qualora il punteggio totale sia inferiore a 18 è necessario ripetere tutte e tre le prove

Per gli studenti frequentanti, è possibile svolgere l'esame in due parti (parziali), una al termine di ciascun modulo del corso



# I contenuti: testi di riferimento

## Testi di riferimento

I lucidi proiettati durante le lezioni in aula o laboratorio sono resi disponibili dai docenti al sito:  
<http://www.unife.it/ing/informazione/fondamenti-informatica>

Al sito indicato, si forniscono anche i testi delle esercitazioni ed alcuni testi d'esame, anche con soluzione.

Argomenti specifici possono essere approfonditi sui seguenti testi.

Per la parte sul linguaggio C:

S. Ceri, D. Mandrioli, L. Sbattella, Informatica: Programmazione, McGraw-Hill, 2006.

A. Bellini, A. Guidi: Guida al Linguaggio C, McGraw-Hill, 1995.

Per la parte sul linguaggio Java, qualsiasi testo introduttivo al linguaggio è adatto.

Si segnalano:

H. M. Dietel, P. J. Dietel: Java Fondamenti di Programmazione, Apogeo.

Lewis John, Loftus William: Java Fondamenti di progettazione software, Addison Wesley.

C. Thomas Wu: Java Fondamenti di Programmazione, Mc Graw Hill.