



UNIVERSITÀ
degli STUDI
di CATANIA

DIPARTIMENTO DI INGEGNERIA ELETTRICA ELETTRONICA E
INFORMATICA

Corso di laurea magistrale in Ingegneria informatica

Anno accademico 2017/2018 - 2° anno

ADVANCED PROGRAMMING LANGUAGES

ING-INF/05 - 9 CFU - 1° semestre

Docente titolare dell'insegnamento

VINCENZA CARCHIOLO

Email: vincenza.carchiolo@unict.it

Edificio / Indirizzo: Edificio 3 - piano V - stanza 11

Telefono: 095 7382359

Orario ricevimento: martedì e venerdì dalle ore 11 alle ore 12

OBIETTIVI FORMATIVI

Gli obiettivi principali sono quello di introdurre i principali costrutti dei linguaggi di programmazione e di fornire gli strumenti necessari per la valutazione critica dei linguaggi di programmazione. Un secondo obiettivo è quello di fornire approfondite conoscenze sulle strutture dei linguaggi di programmazione che consentano di comprendere l'effettiva funzionalità nei diversi campi di applicazione.

Il corso illustra i linguaggi di programmazione fornendo una tassonomia basata sulle applicazioni. Saranno studiati i linguaggi formali e alcuni linguaggi di programmazione orientati ad applicazioni specifiche quali mobile, cloud e Big Data. Saranno studiate le tecniche avanzate dei moderni linguaggi di programmazione con particolare attenzione ai linguaggi multi paradigma, all'uso dei tipi, al type checking dinamico, programmazione concorrente. Tali tecniche saranno analizzate nel contesto dei seguenti linguaggi: Scala, R, Swift, GO.

PREREQUISITI RICHIESTI

Conoscenza delle architetture dei calcolatori base e le relative problematiche, i linguaggi assembly.

Conoscenze di base di Informatica e di programmazione ad oggetti

FREQUENZA LEZIONI

La frequenza non è obbligatoria ma è fortemente consigliata

CONTENUTI DEL CORSO

Il corso è articolato in cinque moduli.

Modulo 1: Caratteristiche dei linguaggi di programmazione e Linguaggi formali per la specifica e la traduzione dei linguaggi di programmazione

Evoluzione dei principali linguaggi di programmazione - Linguaggi formali per la descrizione del lessico della sintassi e della semantica dei linguaggi di programmazione

Tipi di dati - Espressioni e dichiarazioni di assegnazione - Strutture di controllo a livello di dichiarazione - Sottoprogrammi e loro implementazione - Gestione della memoria - Garbage collector - Exception e event handler

Modulo 2; Programmazione funzionale orientata agli oggetti: il linguaggio SCALA

SINTASSI SCALA: Classes and Objects, Basic Types and Operations, Functional Objects Built-in Control Structures, Functions and Closure, Astrazione, Ereditarietà e gerarchia delle classi, Lambda Calculus, Liste, Pattern Matching, Attori e concorrenza, GUI

Modulo 3: Il linguaggio GO

Introduzione al linguaggio Go, Sintassi del linguaggio :Tipi dati e variabili e Costrutti di controllo e decisione, Strutture dati: array, slice e map, Funzioni e defer Struttura della memoria e puntatori Programmazione Object-Oriented, P, Gestione dell'Input e Output su terminale e file, Il compilatore e il garbage collector

Modulo 4: il linguaggio R

Introduzione al Linguaggio R, Sintassi del linguaggio R, arrays, matrici e data frames. Uso e definizione di procedure, funzioni e packages. Vvectorisation, loops, control structures (if, while, for), Ottimizzazione non-lineare e convergenza. I/O e visualizzazione

Modulo 5: Programmazione di sistemi mobili: IL linguaggio Swith

Introduzione al Linguaggio, La sintassi , Tipi di Dato, Variabili, Costanti e Operatori, , Strings e caratteri, Arrays e Dizionari, Enumerazioni, Le strutture di Controllo, Funzioni, Classi, proprietà e metodi, Ereditarietà, Inizializzazione e deinilizzazione, Protocolli, Generic e Controllo degli Accessi

TESTI DI RIFERIMENTO

[[T1] Materiale fornito del docente on line

[T2] Sebesta, Concepts of Programming Languages, 11th Edition - Pearson

[T3] Martin Odersky, Lex Spoon, and Bill Venner: Programming in Scala, Third Edition, Artima

[T4] Alan A. A. Donovan and [Brian W. Kernighan](#), Go Programming Language, Addison-Wesley Professional Computing Series

[T5] Norman Matloff, The Art of R Programming, ISBN-13: 978-1-59327-384-2

ALTRO MATERIALE DIDATTICO

studium.unict.it

PROGRAMMAZIONE DEL CORSO

	* Argomenti	Riferimenti testi
1	* Valutazione dei linguaggi di programmazione	[T2] 1.1-1.4 (inclusive), 1.6
2	Storia dei linguaggi di programmazione	[T2] 2
3	* Descrizione della sintassi e della semantica di un linguaggio di programmazione	[T2]+ capitolo 3
4	* Analisi lessicale e sintattica	[T2] capitolo 4
5	* Naming, Binding e Scope	[T2] capitolo 5
6	* Data Type e Valutazione delle Espressioni	[T2] capitolo 6, 7
7	* Gestione della memoria e Garbage Collection	Aho Ulmann, Compilers: Principles, Techniques, and Tools, 2nd Edition, cap 7.4 -7.8
8	Strutture di controllo e sottoprogrammi	[T2] capitolo 8,9,10
9	Tecniche avanzate di programmazione	[t2] capitolo 13,14
10	* SCALA: Sintassi Classes and Objects, Basic Types and Operations, Functional Objects Built-in Control Structures, Functions and Closures	[T1] CAP1, 2, 3, 4,5,6,7 e 8
11	* SCALA: Astrazione, Ereditarietà e gerarchia delle classi	[T3] CAP da 9 a 13
12	* SCALA: Liste	[T3] cap 16
13	SCALA: Pattern Matching	[T3] cap 15
14	SCALA: Attori e concorrenza	[T3] cap 30
15	SCALA: GUI	[T3] cap 32
16	GO: Sintassi,	
17	Sintassi del Linguaggio R	
18	Sintassi del Linguaggio Swift	

* Conoscenze minime irrinunciabili per il superamento dell'esame.

N.B. La conoscenza degli argomenti contrassegnati con l'asterisco è condizione necessaria ma non sufficiente per il superamento dell'esame. Rispondere in maniera sufficiente o anche più che sufficiente alle domande su tali argomenti non assicura, pertanto, il superamento dell'esame.

VERIFICA DELL'APPRENDIMENTO

MODALITÀ DI VERIFICA DELL'APPRENDIMENTO

E' previsto un esame di tipo orale e la discussione dell'elaborato assegnato dal docente

PROVE IN ITINERE

NON SONO PREVISTE PROVE IN ITINERE

PROVE DI FINE CORSO

NON CI SONO PROVE IN ITINERE NE PROVE SCRITTE DI FINE CORSO

ESEMPI DI DOMANDE E/O ESERCIZI FREQUENTI

Disponibili su Studium
