



Programma del Corso "Linguaggi di Programmazione e Compilatori"

- Codice: F0151
- Tipo di corso: Obbligatorio (Laurea in Informatica percorso Generale)
- Livello del corso: Lauree di Primo Livello
- Semestre: 1

Numero di crediti ECTS: (Laurea in Informatica) 6 (carico 150 ore)

Docenti: Sergio Orefice (Sergio.Orefice@univaq.it)

1	<b>Obiettivi del corso</b>	Nel corso viene illustrata la struttura dei compilatori e sono presentate le principali metodologie impiegate nelle varie fasi del processo di compilazione, in particolare i principi e le tecniche della parte di analisi di un compilatore
2	<b>Contenuti del corso e risultati formativi (descrittori di Dublino)</b>	<p>Gli argomenti trattati nel corso comprendono:</p> <ul style="list-style-type: none"> <li>• Elementi di teoria di linguaggi formali. Introduzione ai compilatori: il modello analisi-sintesi, fasi della compilazione, interpreti</li> <li>• Architettura e ruolo dell'analizzatore lessicale: tokens-patterns-lessemi, buffering dell'input, algoritmo di Thompson</li> <li>• Analisi sintattica: parsing top-down, parsing bottom-up. Parsing predittivo non ricorsivo</li> <li>• Parsing LR: tecniche SLR, LALR, LR canonica</li> <li>• Gerarchia di grammatiche LR e gestione di grammatiche ambigue</li> <li>• Traduzione guidata dalla sintassi. Definizioni dirette dalla sintassi: attributi ereditati e sintetizzati, definizioni con S-attributi, definizioni con L-attributi</li> <li>• Schemi di traslazione. Analisi semantica e type checking</li> <li>• Generazione di codice intermedio. Three-address code: sintassi ed esempi di generazione</li> </ul> <p>Alla fine del corso, lo studente dovrebbe:</p> <ul style="list-style-type: none"> <li>• conoscere la struttura e il ruolo dei compilatori, conoscere e capire i principi e le tecniche utilizzate nelle varie fasi del processo di compilazione</li> <li>• applicare le metodologie studiate allo sviluppo di semplici progetti software riguardanti la costruzione di parti di compilatore</li> <li>• saper valutare le caratteristiche delle componenti lessicali e sintattiche di una specifica di traduzione guidata dalla sintassi e scegliere la soluzione più adatta al suo sviluppo</li> <li>• capire e saper spiegare le relazioni fra strumenti e risultati provenienti dalla teoria dei linguaggi formali e le loro applicazioni nelle fasi di analisi del compilatore</li> <li>• essere capace di leggere e capire altri testi su argomenti correlati</li> </ul>
3	<b>Prerequisiti</b>	Conoscenza delle nozioni base della teoria dei linguaggi formali e della programmazione
4	<b>Modalità e lingua di insegnamento</b>	<p>Lezioni ed esercitazioni</p> <p><b>Lingua:</b> Italiano</p> <p><b>Testi/Bibliografia</b></p> <ul style="list-style-type: none"> <li>• A.V. Aho, R. Sethi, J.D. Ullmann, <i>Compilers, principles, techniques and tools</i>. Addison-Wesley, Reading, Mass.</li> </ul>
5	<b>Metodi di accertamento</b>	<p>Gli studenti sono incoraggiati a partecipare attivamente alle lezioni facendo domande e discutendo le soluzioni adottate negli esempi sviluppati in aula. L'esame consiste in una prova scritta sugli argomenti trattati nel corso. E' prevista anche una prova intermedia scritta che copre la prima parte del programma del corso in modo che gli studenti possano dividere il carico di studio. L'esame scritto (durata 2 ore) è costituito da un insieme di domande per la verifica delle competenze teoriche/formali e per la verifica della capacità di comprendere e risolvere esercizi significativi. Criteri di valutazione saranno: la padronanza dei principi e delle tecniche presentate nel corso, nonché la capacità di applicarle; la chiarezza e la completezza delle spiegazioni.</p>