



Principali informazioni sull'insegnamento

Denominazione dell'insegnamento	Linguaggi di Programmazione (A-L)	
Corso di studio	Informatica e Tecnologia per la Produzione del Software	
Anno Accademico	2023/24	
Crediti formativi universitari (CFU) / European Credit Transfer and Accumulation System (ECTS)	9 CFU	
Settore Scientifico Disciplinare	INF/01	
Lingua di erogazione	Italiano	
Anno di corso	Primo	
Periodo di erogazione	2^ semestre, le date esatte sono riportate nel manifesto/regolamento dell'anno accademico di riferimento	
Obbligo di frequenza	La frequenza è fortemente raccomandata	
Sito web del corso di studio	https://www.uniba.it/it/ricerca/dipartimenti/informatica/didattica/corsi-di-laurea/corsi-di-laurea	

Docente/i	
Nome e cognome	Valeria Carofiglio
Indirizzo mail	valeria.carofiglio@uniba.it
Telefono	080-5443268
Sede	Dipartimento di Informatica, Via Orabona 4, 70125, Bari. Stanza n.561, 5^ piano.
Sede virtuale	Piattaforma e-learning UNIBA - https://elearning.uniba.it/
Sito web del docente	http://www.di.uniba.it/intint/people/valeria.html
Ricevimento (giorni, orari e modalità, es. su appuntamento)	appuntamento da concordare per email con il docente

Syllabus	
Obiettivi formativi	L'insegnamento si propone di introdurre:



	<ul style="list-style-type: none">● i fondamenti teorici dei linguaggi di programmazione;● le tecniche sottostanti la progettazione dei linguaggi di programmazione. <p>Teoria dei linguaggi formali, comprensione dei meccanismi secondo i quali si implementa un linguaggio di programmazione e associazione delle operazioni svolte dal compilatore ai fondamenti teorici dei linguaggi formali.</p>
Prerequisiti	<p>Le seguenti conoscenze preliminari facilitano ed accelerano la comprensione degli argomenti dell'insegnamento:</p> <ul style="list-style-type: none">● da Programmazione: basi della programmazione imperativa, ricorsione, processo di compilazione;● da Matematica Discreta: teoria degli insiemi, relazioni, principio di induzione, strutture algebriche elementari, dimostrazioni di tipo induttivo e deduttivo, definizione di funzione
Contenuti di insegnamento (Programma)	<p>La numerazione degli argomenti ne indica l'ordine di presentazione e di studio. Inoltre, l'indice di ogni argomento è utile per ritrovare i relativi contenuti nel testo di riferimento. La suddivisione in prima e seconda parte è semplicemente per facilitare lo studente nella comprensione della modalità d'esame, illustrata nel seguito.</p> <p>Prima Parte</p> <ol style="list-style-type: none">1. Introduzione ai linguaggi di programmazione ed ai linguaggi formali: La nozione di macchina astratta e l'interprete. Interprete. Un esempio di macchina astratta: la macchina hardware. Implementazione di un linguaggio. Realizzazione di una macchina astratta. Implementazione: il caso ideale. Implementazione: il caso reale e la macchina intermedia. Gerarchie di macchine astratte.2. Descrivere un linguaggio di programmazione Livelli di descrizione. Grammatica e sintassi. Vincoli sintattici contestuali. Compilatori. Semantica. Pragmatica. Implementazione.3. Linguaggi regolari per l'analisi lessicale: Token. Linguaggi formali e operazioni. Espressioni regolari. Automi finiti: DFA e NFA. equivalenza. Da espressioni regolari ad automi finiti. Automi finiti e grammatiche. Algoritmo di Hopcroft (fornito dal docente, fuori dal testo di riferimento). Minimizzare un DFA. Generatori di analizzatori lessicali. Dimostrare che un linguaggio non è regolare.4. Linguaggi liberi da contesto per l'analisi sintattica: Linguaggi, derivazioni e alberi. Automi a pila. Dimostrare che un linguaggio non è libero. Oltre i linguaggi liberi. Analizzatori sintattici. Parser topdown. Parser a discesa ricorsiva. First e Follow. Grammatiche LL(1). Parser LL(1) non ricorsivi. Grammatiche LL(k). Parser bottom-up (cenni). Linguaggi e grammatiche deterministici. Generatori di analizzatori sintattici. <p>Seconda parte</p> <ol style="list-style-type: none">5. I nomi e l'ambiente: Nomi e oggetti denotabili. Ambiente e blocchi. Tipi di ambiente. Operazioni sull'ambiente. Regole di scope. Scope statico e dinamico.6. Gestione della memoria: Gestione statica e dinamica della memoria. Gestione dinamica mediante pila. Record di attivazione per i blocchi in-line e le procedure. Gestione della pila. Gestione dinamica mediante heap. Blocchi di dimensione fissa e variabile. Implementazione delle regole di scope statico (mediante catena statica e vettore del displ) e dinamico (mediante lista di associazioni e CRT).7. Cenni alla sintassi, semantica e implementazione dei principali costrutti dei linguaggi di programmazione imperativi:



	Strutturare il controllo (Sintassi, semantica e valutazione delle espressioni, del comando di assegnazione, dei comandi per il controllo di sequenza espliciti, condizionale, iterativo, ricorsivo e ricorsivo in coda). Astrarre sul controllo (Sottoprogrammi, funzioni e passaggi di parametri). Strutturare i dati (Tipi di dati come supporto all'organizzazione concettuale e per la correttezza. Sistemi e implementazione di tipi noti, controlli statici e dinamici, equivalenza, compatibilità e conversione di tipi. Cenni di Controllo e inferenza di tipo). Astrarre sui dati (Tipi di dati astratti, Nascondere l'informazione. Indipendenza dalla rappresentazione. Moduli		
Testi di riferimento	<p>[Gabrielli] M. Gabrielli, S. Martini. Linguaggi di Programmazione, Principi e paradigmi. 2/ed., McGraw-Hill (2011).</p> <p>Gli studenti che lo desiderano possono ottenere i testi in prestito dalla Biblioteca. Può convenire verificarne la disponibilità mediante il Sistema Bibliotecario di Ateneo https://opac.uniba.it/easyweb/w8018/index.php? e contattare la biblioteca per concordare il prestito.</p>		
Note ai testi di riferimento	<p>Lo studente tenga in considerazione che gli argomenti del programma sono ampiamente trattati nel libro di testo adottato, e che gli stessi compaiono in tale testo nel medesimo ordine e con la medesima titolazione in cui compaiono nel programma.</p> <ol style="list-style-type: none"> 1. <i>Introduzione ai linguaggi di programmazione e ai linguaggi formali:</i> Capitolo 1 2. <i>Descrivere un linguaggio di programmazione:</i> Capitolo 2 3. <i>Linguaggi regolari per l'analisi lessicale:</i> Capitolo 3; Algoritmo di Hopcroft (ADA) 4. <i>Linguaggi liberi da contesto per l'analisi sintattica:</i> Capitolo 4 5. <i>I nomi e l'ambiente:</i> Capitolo 6 6. <i>Gestione della memoria:</i> Capitolo 7 7. <i>Cenni alla sintassi, semantica e implementazione dei principali costrutti dei linguaggi di programmazione imperativi:</i> Capitoli 8, 9, 10, 11 <p>Si precisa che non saranno fornite le slide del corso.</p> <p>Inoltre all'indirizzo https://www.mheducation.it/linguaggi-di-programmazione-2-ed-9788838665738-italy, nella sezione risorse e download, area studenti, possono essere reperiti tutti gli approfondimenti richiamati nel libro di testo adottato.</p> <p>Infine, sulla piattaforma di e-learning ADA https://elearning.di.uniba.it saranno reperibili esercizi ed esempi delle prove d'esame.</p>		
Organizzazione della didattica			
Ore			
Totali	Didattica frontale	Esercitazioni	Studio individuale
225 ore	56 ore	30 ore	139 ore
CFU/ETCS			
9 CFU	7 CFU	2 CFU	



Metodi didattici	
	<ul style="list-style-type: none">- Lezioni frontali condotte con l'ausilio di dispense proiettate in aula- Svolgimento in aula di due tipologie di esercitazione: (a) esercizi svolti interamente dal docente con indicazione delle soluzioni; (b) esercitazioni guidate in cui gli studenti risolvono insieme al docente quesiti simili a quelli da affrontare nelle prove d'esame. <p>Entrambe le tipologie di esercitazione sono svolte con l'obiettivo di acquisire dimestichezza con gli argomenti svolti nel corso</p>
Risultati di apprendimento previsti	
Conoscenza e capacità di comprensione	Conoscenza delle caratteristiche dei linguaggi di programmazione dal punto di vista sintattico, semantico, implementativo e pragmatico, principalmente nel contesto del paradigma imperativo.
Conoscenza e capacità di comprensione applicate	Competenze relative alla progettazione, implementazione e funzionamento dei compilatori per i moderni linguaggi di programmazione
Competenze trasversali	<p>Autonomia di giudizio</p> <p>Capacità di valutazione nell'ambito dell'utilizzo dei linguaggi di programmazione, tenendo conto delle loro caratteristiche intrinseche ed esterne</p> <p>Abilità comunicative</p> <p>Illustrazione delle caratteristiche tecniche degli strumenti e delle metodologie informatiche relative ai linguaggi di programmazione.</p> <p>Capacità di apprendere in modo autonomo</p> <p>Sviluppo delle capacità di apprendere e di orientarsi agilmente nelle problematiche relative alla comprensione e all'utilizzo delle tecnologie informatiche nel suo specifico campo di applicazione.</p>
Valutazione	
Modalità di verifica dell'apprendimento	<p>Prova d'esame L'esame consiste in una prova scritta ed una orale (non obbligatoria a discrezione del docente).</p> <p>Prova Scritta nella quale si richiede di rispondere alle seguenti tipologie di quesiti:</p> <ul style="list-style-type: none">● svolgimento di esercizi sulla teoria dei linguaggi formali;



- svolgimento di esercizi sulle regole di scope
- enunciazione di definizioni sulla teoria dei linguaggi formali;
- quesiti aperti sulla seconda parte del corso

Regolamentazione della prova d'esame:

- Obbligo di prenotazione per tempo su esstre. In nessun caso, senza prenotazione su esstre, è possibile sostenere la prova.
- durata della prova: varia tra 60 e 90 minuti, in base alla tipologia di quesiti da risolvere;
- nessun materiale è consultabile durante lo svolgimento della prova
- la valutazione è espressa in venticinquesimi. Tale prova si ritiene superata con una votazione non inferiore a 18/25
- comunicazione dei risultati: piattaforma esse3.

Prova Orale

In caso di prova scritta superata (con votazione minima 18/25), vi è una prova orale non obbligatoria a **discrezione del docente**, riguardante la seconda parte del programma. Il superamento della prova puo' dare luogo a premialità per un massimo di 6 punti.

La valutazione finale sarà ottenuta dalla somma delle valutazioni conseguite nella prova scritta e quella orale

Prova in Itinere

Durante lo svolgimento del corso, ed in ogni caso al termine della prima parte del programma, con un preavviso di massimo una settimana, è prevista una prova scritta, **NON** obbligatoria, rivolta ai frequentanti, riguardante la prima parte del programma. L'obiettivo della prova in itinere è quello di incoraggiare lo studente ed uno studio regolare e costante "giorno per giorno", oltre che di fornire un riscontro allo studente sul proprio livello di comprensione degli argomenti dell'insegnamento.

Si richiede:

- esecuzione di esercizi sulla teoria dei linguaggi formali.

Regolamentazione della prova in Itinere:

- Obbligo di prenotazione per tempo su esstre. In nessun caso, senza prenotazione su esstre, è possibile sostenere la prova.
- durata della prova: 75 minuti;
- materiale consultabile: nessuno;
- La valutazione è espressa in venticinquesimi e si considera superata con una votazione minima di 16/25
- Il superamento della prova in itinere esonera lo studente dallo svolgimento della prova scritta per l'intero anno accademico, obbligandolo però ad una prova orale in una delle date possibili per l'anno accademico in corso, che verta sugli argomenti affrontati durante il corso, dopo la data della prova in itinere.
- La prova in itinere dà una premialità di massimo 25 punti, e perde di validità in caso di non superamento della prova orale programmata dallo studente. In tal caso, lo studente è obbligato a svolgere una ordinaria prova d'esame in una delle date possibili (come se la prova in itinere non fosse stata mai sostenuta).
- la valutazione finale sarà ottenuta dalla somma delle valutazioni conseguite nella prova in itinere ed in quella orale.
- comunicazione dei risultati: piattaforma esse3

Criteri di valutazione

- *Conoscenza e capacità di comprensione:*



	<ul style="list-style-type: none">- Capacità di enunciare con rigore definizioni e teoremi della teoria dei linguaggi formali;- Capacità di descrivere con precisione e chiarezza le tecniche ed i fondamenti teorici alla base dei meccanismi di analisi e traduzione dei linguaggi di programmazione;- Capacità di descrivere con chiarezza le caratteristiche più rilevanti e comuni ai linguaggi appartenenti alla classe dei linguaggi imperativi.- <i>Conoscenza e capacità di comprensione applicate:</i><ul style="list-style-type: none">- Capacità di classificare un linguaggio formale, mediante gli strumenti appresi;- Capacità di generare, descrivere e riconoscere un linguaggio formale, mediante la costruzione di grammatiche, la definizione di espressioni regolari, e la progettazione di opportune macchine.- Capacità di descrivere in pseudolinguaggio ed efficacemente gli aspetti più rilevanti della classe dei linguaggi imperativi, attraverso anche esercizi pratici.- <i>Autonomia di giudizio:</i><ul style="list-style-type: none">- Modalità adottate per la verifica delle soluzioni proposte- <i>Abilità comunicative:</i><ul style="list-style-type: none">- Chiarezza nella descrizione delle soluzioni proposte ai quesiti.- <i>Capacità di apprendere:</i><ul style="list-style-type: none">- Capacità di astrazione, di ragionamento per analogia e dimostrazione di creatività nella risoluzione dei quesiti.
<p>Criteria di misurazione dell'apprendimento e di attribuzione del voto finale</p>	<p><u>Prova Scritta</u> Ad ogni quesito da svolgere sarà attribuito un punteggio, assicurando che la somma dei punteggi sia pari a 25 La determinazione del voto attribuito alla risposta data ad ogni quesito tiene conto dei seguenti criteri:</p> <ol style="list-style-type: none">1) la correttezza della soluzione, della dimostrazione formale o della definizione fornita;2) la completezza della soluzione, della dimostrazione formale o della definizione fornita;3) la logica seguita dallo studente nel proporre la soluzione o la dimostrazione formale;4) l'utilizzo di un adeguato formalismo per la descrizione della soluzione, della dimostrazione o della definizione. <p>Per ottenere una valutazione sufficiente, le risposte fornite ai quesiti devono totalizzare almeno 18.</p> <p><u>Prova Orale</u> Sulla seconda parte del corso. È richiesta competenza e chiarezza nell'esposizione degli argomenti, oltre che capacità di argomentazioni coerenti, collegamenti e relazioni tra concetti.</p> <p>Il superamento della prova può dare luogo a premialità per un massimo di 6 punti. L'esito della prova orale concorre alla valutazione finale come descritto nel riquadro "Modalità di verifica dell'apprendimento".</p> <p><u>Prova in itinere</u> La valutazione della prova è espressa in venticinquesimi. La prova in itinere si intende superata quando il voto conseguito è maggiore o uguale a 16. L'esito della prova in itinere concorre alla valutazione finale come descritto nel riquadro "Modalità di verifica dell'apprendimento".</p> <p><u>Prova d'esame</u> La valutazione della prova è espressa in trentesimi.</p>



	<p>La prova d'esame si intende superata quando il voto conseguito è maggiore o uguale a 18.</p> <p>Attribuzione della lode: la lode è attribuita quando la logica seguita dallo studente nel proporre la soluzione o la dimostrazione formale evidenzia particolari capacità di astrazione, ragionamento per analogia, creatività.</p>
Altro	<p>E' fortemente consigliato seguire le lezioni in presenza, di modo da essere stimolati ad uno studio regolare e costante "giorno per giorno". Si suggerisce, inoltre di affidarsi esclusivamente alle informazioni/comunicazioni fornite sui siti ufficiali del Dipartimento di Informatica, ovvero sui gruppi social solo se costituiti e amministrati esclusivamente dai docenti dei relativi insegnamenti:</p> <ul style="list-style-type: none">• https://www.uniba.it/it/ricerca/dipartimenti/informatica/didattica/corsi-di-laurea/corsi-di-laurea• https://www.uniba.it/it/ricerca/dipartimenti/informatica• https://elearning.uniba.it/ <p>I programmi degli insegnamenti sono disponibili qui:</p> <ul style="list-style-type: none">• https://elearning.uniba.it/course/index.php?categoryid=285 <p>Le informazioni che tutti gli studenti dovrebbero conoscere sono scritte nei Regolamenti didattici e manifesti degli studi disponibili nel sito:</p> <ul style="list-style-type: none">• h https://www.uniba.it/it/ricerca/dipartimenti/informatica/didattica/corsi-di-laurea/corsi-di-laurea <p><i>Si suggerisce agli studenti di diffidare dalle informazioni e dei materiali circolanti su siti o gruppi social non ufficiali, poiché spesso sono risultati non affidabili, non corretti o incompleti. Per ogni dubbio, fissare un incontro con il docente secondo le modalità previste per il ricevimento.</i></p> <ul style="list-style-type: none">• Link all'insegnamento sulla piattaforma e-learning del dipartimento ADA: https://elearning.uniba.it/ <p><u>Suggerimenti per una corretta preparazione, in particolare per studenti non frequentanti:</u></p> <ol style="list-style-type: none">1) studiare gli argomenti teorici nell'ordine suggerito, svolgendo gli esercizi che sono indicati alla fine di ogni capitolo.2) L'ordine con il quale sono presentati gli argomenti è fondamentale per la comprensione degli stessi. Tentare di risolvere gli esercizi apprendendo solo dalle tracce di esempio e relative soluzioni non consente di acquisire le basi culturali per poter superare l'esame. Comprendere perché si arriva ad una certa soluzione è molto più importante che svolgere molti esercizi senza comprenderne del tutto le basi teoriche.3) Svolgere gli esercizi, dei quali sono date le soluzioni, disponibili su ADA, in modo da confrontare le proprie soluzioni con quelle fornite dal docente.4) Svolgere gli esercizi tratti da prove d'esame per valutare la propria preparazione.5) Contattare il docente per chiedere eventuali chiarimenti e/o verificare le proprie soluzioni agli esercizi svolti in autonomia prima di iscriversi alla prova d'esame.