

Linguaggi di Programmazione e Laboratorio

Domande di teoria

Che cos'è una macchina astratta per un linguaggio di programmazione L?

Si descriva l'implementazione interpretativa pura di un linguaggio.

Si descriva l'implementazione compilativa pura di un linguaggio.

Si descriva l'implementazione di un linguaggio mediante macchina intermedia.

Che cosa sono la grammatica, la semantica e la pragmatica di un linguaggio?

Che cosa sono le grammatiche libere?

Si descriva la notazione di Backus e Naur di una grammatica libera.

Che cosa sono le derivazioni e gli alberi di derivazione di una grammatica libera?

Si dia la definizione di grammatica ambigua.

Che cosa sono i vincoli sintattici contestuali?

Si descrivano i vari passi di un compilatore.

Che cosa sono le espressioni regolari?

Che cosa sono gli automi a stati finiti?

In che cosa si differenziano gli automi finiti non-deterministici e deterministici?

Come si passa da un NFA a un DFA?

Come si costruisce un NFA da un'espressione regolare?

Quando una grammatica libera è regolare?

Quali sono le relazioni tra espressioni regolari, NFA, DFA e grammatiche regolari?

Si dia la definizione di automa a pila non deterministico e di linguaggi accettati dall'automa.

Si dia la definizione di linguaggio libero da contesto, PDA deterministico, di linguaggio libero deterministico e si descrivano le relazioni tra di essi.

Si dia la definizione di grammatica dipendente dal contesto e di grammatica ricorsivamente enumerabile.

Si descriva la gerarchia delle grammatiche di Chomski e si indichi quali macchine riconoscono ciascun tipo di grammatica.

Si dia un esempio di linguaggio libero non regolare e di linguaggio contestuale non libero.

Che cosa fa l'analizzatore sintattico di un linguaggio e si spieghino le distinzioni tra analizzatori deterministici e non deterministici e tra analizzatori top-down e bottom-up.

Si illustri a grandi linee come si possono eliminare i simboli inutili dalle grammatiche.

Si illustri a grandi linee come eliminare la ricorsione sinistra dalle grammatiche.

Si definisca un parser top-down a discesa ricorsiva.

Che cos'è la tabella di parsing?

Quando una grammatica è LL(1) e che cosa fa un parser top-down a discesa ricorsiva per tali grammatiche?

Si dia l'algoritmo per un parser top-down LL(1) non ricorsivo.

Si dia l'algoritmo per un parser bottom-up.

Si definiscano a grandi linee i parser LL(k), LR(k), SLR(1) e LALR(1).

Si definiscano le grammatiche LL(k) e LR(k) e si dica che relazione c'è tra di loro.

Che cos'è la programmazione logica?

Si dia la definizione di termine logico.

Si dia la definizione di atomo logico.

Si dia la definizione di formula logica ben formata.

Si dia la definizione di programma logico.

Si dia la definizione di clausola logica.

Che differenza c'è tra le variabili nella programmazione imperativa e in quella logica?

Si dia la definizione di sostituzione.

Si dia la definizione di Most General Unifier.

Che cosa fa l'algoritmo di unificazione?

Si descriva l'interpretazione dichiarativa di un programma logico.

Si descriva l'interpretazione procedurale di un programma logico.

Si definisca la risoluzione SLD.

Come funziona il backtracking in Prolog?

Si fornisca il codice in Prolog del predicato member che verifica l'appartenenza di un elemento ad una lista.

Si fornisca il codice in Prolog del predicato append che concatena due liste.

Come viene interpretata in Prolog la negazione?

Che cosa sono i nomi e l'ambiente (environment)?

Che cosa sono le regole di visibilità?

Si spieghi la gestione della memoria dinamica tramite stack e record di attivazione.

Si illustrino le operazioni che sono eseguite quando viene eseguita una procedura.

Si spieghino le possibili strategie per la gestione della memoria dinamica tramite heap.

Si illustrino la differenza tra linguaggi imperativi e dichiarativi.

Si parli del goto.

Si parli dei comandi condizionali.

Si parli dei comandi iterativi.

Che cos'è la programmazione strutturata?

Che cos'è la tail recursion?

Che differenza c'è tra ricorsione e iterazione?

Quando un linguaggio ha first-class functions?

Che cosa fa il garbage collector e perché è utile?

Si illustrino le due tecniche principali per la garbage collection.

Come sono implementati gli oggetti?

Come si svolge il dynamic method lookup?

Come sono rappresentate le classi?

Come si invocano i metodi nel caso di ereditarietà singola?