# Linguaggi di Programmazione

Domande teoria

**LOGICA MATEMATICA**

1. Spiegare come avviene la dimostrazione per assurdo mediante uso del principio di risoluzione

La **dimostrazione per assurdo** è un metodo di prova in cui si assume che una certa affermazione sia falsa e si cerca di derivare una contraddizione da questa assunzione.

Se si arriva a una contraddizione, si conclude che l’affermazione iniziale deve essere per forza vera.

Il principio di risoluzione è una regola di inferenza utilizzata per dedurre conclusioni a partire da un insieme di clausole. Essa opera su formule ben formate (FBF) trasformate in **forma normale congiunta**.

Processo di dimostrazione

1. **Assunzione iniziale**

Si inizia assumendo che l’affermazione che si desidera dimostrare sia falsa. Quindi, si assume NOT(P).

1. **Rappresentazione in forma clausale**

Si traduce l’affermazione NOT(P) e tutte le altre premesse in forma clausale. Le clausole sono espressione logiche che contengono disgiunzioni letterali.

1. **Applicazione principio di risoluzione**

Si applica il principio di risoluzione alle clausole ottenute. Si cercano coppie di clausole che contengono letterali opposti. Quando si trova una coppia, si risolvono per ottenere una nuova clausola.

1. **Iterazione**

Si continua ad applicare il principio di risoluzione, generando nuove clausole fino a quando si ottiene la clausola vuota o si esauriscono le possibilità.

1. **Conclusione**

Se si ottiene una clausola vuota, ciò implica che l’assunzione NOT(P) sia falsa. Pertanto, si conclude che P deve essere vera.

1. Cos’è un calcolo logico

Un **calcolo logico** è un sistema formale che utilizza simboli e regole per rappresentare e manipolare proposizioni logiche.

Un calcolo logico garantisce che tutte le nuove formule generate siano vere se l’insieme di assiomi consiste solo di formule vere. Questo processo di generazione si chiama **dimostrazione.**Esistono diversi tipi di calcolo logico:

**Calcolo Proposizionale**

Si occupa di proposizione che possono essere vere o false e delle loro combinazioni tramite operatori logici.

**Calcolo del primo ordine**

Estende il calcolo proposizionale introducendo quantificatori (come “per ogni” ed “esiste”) e predicati, che permettono di esprimere affermazioni più complesse riguardanti oggetti e le loro proprietà.

1. Dare una definizione di regola di inferenza, e fare un esempio di almeno due regole di inferenza viste a lezione.

L’inferenza è un processo logico che consente di derivare conclusioni a partire da premesse. In altre parole, è una regola che stabilisce come si possono ottenere nuove conclusioni da premesse già note. Le regole sono fondamentali nella logica formale perché permettono di dimostrare teoremi.

Alcuni esempi comuni di regole di inferenza:

1. **Modus Ponens**: Se abbiamo "Se P allora Q" (P → Q) e "P", possiamo concludere "Q".
2. **Modus Tollens**: Se abbiamo "Se P allora Q" (P → Q) e "Non Q", possiamo concludere "Non P".
3. **Disgiunzione**: Se abbiamo "P o Q" (P ∨ Q) e "Non P", possiamo concludere "Q".

**PROLOG**

1. Cos’è il cut? Spiegarlo in termini di backtracking sugli alberi di derivazione

Il cut è predicato speciale per controllare la sequenza che l’interprete Prolog va a considerare nella risoluzione di un goal.

Considerando una clausola generica

C = a :- b1, b2, ..n bk, !, bk+1, …, bn.

L’effetto del cut è il seguente

Se il goal corrente G unifica con a e b1, .., bk hanno successo allora il dimostratore si impegna inderogabilmente alla scelta di C per dimostrare G.

Ogni clausola alternativa (successiva) per a che unifica con G viene ignorata.

Se qualche bj con j > k fallisse, il backtracking si fermerebbe al cut !.

Quando il backtracking raggiunge il cut, allora il cut fallisce e la ricerca procede dall’ultimo punto scelto prima che G scegliesse C.