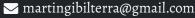
# Esempi di prove in itinere

Domande a risposta multipla sugli argomenti del corso

Tutorato di Fondamenti di Informatica 19/04/2024

#### Martin Gibilterra

Università di Catania



github.com/w8floosh

in linkedin.com/in/w8floosh

Domande sui sistemi formali e sulla logica

proposizionale



Una derivazione senza ipotesi:

- A. è un'ipotesi
- B. è una qualsiasi sequenza di assiomi e conclusioni di regole di inferenza
- C. è una qualsiasi sequenza di fbf
- D. non è soddisfacibile

Se  $\overline{B}(a) = 1$  e  $\overline{B}(b) = 0$ , allora è falso affermare che:

- A.  $a \Rightarrow b$  sia una tautologia
- B.  $a \Rightarrow b$  sia contraddittoria
- C.  $\neg(a \Rightarrow b)$  sia soddisfacibile
- D.  $\neg(a \Rightarrow \neg(b))$  sia contraddittoria

L'insieme delle variabili proposizionali è:

- A. non numerabile
- B. infinito
- C. finito
- D. indefinito

La tabella di verità che mostra la soddisfacibilità di  $(p \Rightarrow (q \Rightarrow r)) \Rightarrow q$  deve essere composta da:

- A. 22 righe
- B. 2<sup>2</sup> colonne
- C. 2<sup>3</sup> colonne
- D. 2<sup>3</sup> righe

Il teorema di correttezza afferma quanto segue: Siano  $\Gamma$  un insieme di fbf in  $P_0$  e  $\alpha$  una fbf in  $P_0$ . Si ha che:

- A. se  $\Gamma \not\vdash_{P_o} \alpha$  allora  $\vdash_{P_o} \alpha$
- B. se  $\Gamma \vdash_{P_o} \alpha$  allora  $\Gamma \vDash \alpha$
- C. se  $\Gamma$  è soddisfacibile allora  $\Gamma \cup \{\alpha\}$  è consistente
- D. se  $\overline{B}(\Gamma) = 1$  allora  $\Gamma \vDash \alpha$



Data una fbf a di  $P_{\circ}$ , il problema di affermare se a è un teorema di  $P_{\circ}$  è:

- A. decidibile
- B. indefinito
- C. insoddisfacibile
- D. soddisfacibile

Domande su automi a stati finiti e linguaggi regolari

Dato l'automa a stati finiti deterministico  $A = < \Sigma$ , Q,  $\delta$ ,  $q_0$ , F >, se  $F = \emptyset$ :

- A. la computazione non termina mai
- B. la computazione termina sempre riconoscendo almeno una stringa
- C. la computazione termina sempre non riconoscendo nessuna stringa
- D. la computazione termina sempre riconoscendo ogni stringa

Dato l'automa a stati finiti deterministico  $A = < \Sigma, Q, \delta, q_o, F >$ , se  $q_o \notin Q$ :

- A. la computazione termina al primo carattere letto
- B. l'automa è definito in modo errato
- C. l'automa non terminerà mai la sua computazione
- D. l'automa termina ogni computazione su uno stato *f* in *F*.

Dato un automa a stati finiti deterministico su un alfabeto di input  $\Sigma = \{a, b\}$ , il diagramma degli stati:

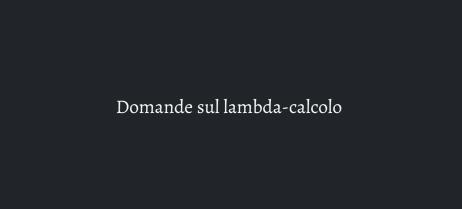
- A. può avere uno stato  $q_0$  con un arco uscente verso uno stato  $q_1$  e viceversa
- B. può avere uno stato  $q_2$  con tre archi uscenti etichettati con a
- C. può avere stati con nessun arco entrante
- D. è sempre un grafo completo

Dato un automa a stati finiti non deterministico valido su un alfabeto di input  $\Sigma = \{a, b\}$ , si consideri la sua tabella di transizione, con  $q_F$  unico stato finale.

	а	b
qo		$\{q_1\}$
$q_1$	$\{q_1\}$	$\{q_F\}$
$q_F$		$\{q_{\rm o}\}$

Cosa si può dire su questo automa?

- A. ha almeno uno stato con nessun arco entrante
- B. la funzione di transizione è completamente definita
- C. non è un automa a stati finiti non deterministico
- D. non presenta transizioni non deterministiche



Quale tra queste non è una componente fondamentale del modello computazionale del  $\lambda$ -calcolo?

- A. assegnare un nome alle espressioni
- B. le variabili
- C. l'applicazione di funzioni
- D. l'operatore di astrazione funzionale unaria  $n \Rightarrow \dots$

Nel lambda-calcolo, dati gli insiemi FV(X) e  $BV(\lambda n.Y)$  rispettivamente l'insieme delle variabili libere del lambda-termine X e l'insieme delle variabili legate del termine  $\lambda n.Y$ , la beta-riduzione è applicabile se e solo se:

A. 
$$FV(X) \cap BV(\lambda n.Y) = \emptyset$$

B. 
$$FV(X) \cap BV(\lambda n.Y) = FV(X)$$

C. 
$$FV(X) = \emptyset$$

D. 
$$BV(\lambda n.Y) = \emptyset$$

Nel lambda calcolo, un termine che non contiene redex si definisce:

- A. fortemente normalizzabile
- B. beta-redex
- C. in forma normale
- D. riducibile

Nel lambda calcolo, la beta-espansione rende possibile:

- A. calcolare funzioni altrimenti incalcolabili
- B. calcolare funzioni su infinito input
- C. calcolare funzioni simulando il modello di Turing
- D. calcolare funzioni ricorsive

Nel lambda calcolo, il termine  $\lambda x.\lambda y.\lambda z.((xx)(\lambda y.zy)(\lambda z.zxy))$  è:

- A. un termine fortemente normalizzabile
- B. una funzione che calcola il fattoriale di x
- C. una funzione curryficata
- D. la funzione somma di Church

Dato un carattere non terminale *X*, quale di queste grammatiche genera un qualsiasi lambda termine?

- A.  $\Lambda ::= X$
- B.  $\Lambda ::= X | \Lambda \Lambda$
- C.  $\Lambda := X | \Lambda \Lambda | XY$
- D.  $\Lambda ::= X | \Lambda \Lambda | \lambda . \Lambda$

Si osservi la seguente alpha-conversione:  $(\lambda x.yx)(\lambda y.zy) \Rightarrow_{\alpha} (\lambda x.zx)(\lambda z.zz)$  Cosa si può affermare con certezza sull'alpha-conversione in questione?

- A. non è corretta perché trasforma la variabile x da legata a libera
- B. non è corretta perché trasforma la variabile z da libera a legata
- C. è corretta
- D. produce due termini con lo stesso significato