

Cognome \_\_\_\_\_ Nome \_\_\_\_\_

Matricola \_\_\_\_\_ Numero di CFU \_\_\_\_\_ Fila 1

**Università degli Studi di Bologna, Corso di Laurea in Informatica  
Esame di LOGICA PER L'INFORMATICA (9 CFU), 16/02/2024**

*Utilizzare i riquadri bianchi per le risposte. Se strettamente necessario, si può allegare un foglio protocollo in coda con ulteriore testo, indicando in alto nome, cognome, fila e matricola.*

L1 (5 punti). Definiamo in teoria degli insiemi il seguente predicato di overlap fra due insiemi:

$$A \not\subseteq B := \exists X.(X \in A \wedge X \in B)$$

Dimostrare in teoria degli insiemi che  $\forall A.((\exists C.A \not\subseteq C) \Rightarrow \exists B.A \not\subseteq B)$ . Ogni passaggio deve abbreviare uno o più passi di deduzione naturale intuizionista al prim'ordine.

Explicitare l'enunciato di tutti gli assiomi utilizzati nella dimostrazione.

Assioma dell'insieme vuoto:  $\forall X.X \notin \emptyset$

Teorema:  $\forall A.((\exists C.A \not\subseteq C) \Rightarrow \exists B.A \not\subseteq B)$

Dimostrazione: sia  $A$  insieme t.c.  $\exists C.A \not\subseteq C$ . Quindi sia  $C$  l'insieme t.c.  $A \not\subseteq C$  o, equivalentemente,  $\exists X.(X \in A \wedge X \in C)$ . Quindi sia  $X$  l'insieme t.c.  $X \in A$  (H1) e  $X \in C$  (H2). Dimostriamo  $\exists B.A \not\subseteq B$  scegliendo l'insieme vuoto per  $B$  e riducendoci a dimostrare  $A \not\subseteq \emptyset$ . Supponiamo  $A \subseteq \emptyset$ , ovvero  $\forall Y.(Y \in A \Rightarrow Y \in \emptyset)$ . Quindi, per H1,  $X \in \emptyset$ . Quindi, per l'assioma dell'insieme vuoto, assurdo. Qed.

L2 (5 punti). Dimostrare in deduzione naturale per la logica al prim'ordine il seguente enunciato. Preferire una prova intuizionista a una classica ove possibile.

$$\forall n.\forall m.(P(f(n), m) \Rightarrow P(n, f(m))), \quad \forall n.\exists m.P(f(n), f(m)) \vdash \exists z.P(m, f(z))$$

$$\frac{\frac{\frac{\forall n.\exists m.P(f(n), f(m))}{\exists x.P(f(m), f(x))} \forall_e[m/n]}{\exists z.P(m, f(z))} \forall_e[m/n] \quad \frac{\frac{\frac{\frac{\frac{\forall n.\forall m.P(f(n), m) \Rightarrow P(n, f(m))}{\forall x.P(f(m), x) \Rightarrow P(m, f(x))} \forall_e[m/n]}{P(f(m), f(y)) \Rightarrow P(m, f(f(y)))} \forall_e[f(y)/x]}{[P(f(m), f(y))]} \Rightarrow_e}{\frac{P(m, f(f(y)))}{\exists z.P(m, f(z))} \exists_i[f(y)/z]} \exists_e[y/x] \quad \Rightarrow_e}{\exists z.P(m, f(z))} \Rightarrow_e$$

Cognome \_\_\_\_\_ Nome \_\_\_\_\_

Matricola \_\_\_\_\_ Numero di CFU \_\_\_\_\_ Fila 1

**Università degli Studi di Bologna, Corso di Laurea in Informatica  
Esame di LOGICA PER L'INFORMATICA (9 CFU), 16/02/2024**

Utilizzare i riquadri bianchi per le risposte. Se strettamente necessario, si può allegare un foglio protocollo in coda con ulteriore testo, indicando in alto nome, cognome, fila e matricola.

L3 (5 punti). Si consideri il seguente ragionamento:

Se l'amore non è per sempre allora la coppia si lascerà o diventerà una coppia aperta. Andrà a finire male e non ci sarà un matrimonio, se la coppia si lascerà. Se diventerà una coppia aperta allora ci sarà un matrimonio e l'amore sarà per sempre. Quindi l'amore sarà per sempre o andrà a finire male.

Verificare la correttezza del ragionamento utilizzando la deduzione naturale per la logica proposizionale. Preferire una prova intuizionista se possibile.

$\neg A \Rightarrow B \vee C, \quad B \Rightarrow D \wedge \neg E, \quad C \Rightarrow E \wedge A \vdash A \vee D$

$$\begin{array}{c}
 \vdots \\
 \hline
 A \vee \neg A \quad EM \\
 \hline
 \frac{\neg A \Rightarrow B \vee C \quad [\neg A] \Rightarrow_e}{B \vee C} \Rightarrow_e \quad \frac{\frac{B \Rightarrow D \wedge \neg E \quad [B] \Rightarrow_e}{D \wedge \neg E} \wedge_{e1} \quad \frac{C \Rightarrow E \wedge A \quad [C] \Rightarrow_e}{E \wedge A} \wedge_{e1}}{\frac{D}{A \vee D} \vee_{i2} \quad \frac{E \wedge A}{A} \vee_{i1}} \Rightarrow_e \quad \frac{A}{A \vee D} \vee_e \\
 \hline
 A \vee D \quad \vee_{e1} \\
 \hline
 A \vee D \quad \vee_e
 \end{array}$$



Cognome \_\_\_\_\_ Nome \_\_\_\_\_

Matricola \_\_\_\_\_ Numero di CFU \_\_\_\_\_ Fila 1

**Università degli Studi di Bologna, Corso di Laurea in Informatica  
Esame di LOGICA PER L'INFORMATICA (9 CFU), 16/02/2024**

*Utilizzare i riquadri bianchi per le risposte. Se strettamente necessario, si può allegare un foglio protocollo in coda con ulteriore testo, indicando in alto nome, cognome, fila e matricola.*

L4 (5 punti). Considerate la seguente sintassi per alberi binari i cui nodi interni sono etichettati con numeri interi:  $T ::= \emptyset \mid \langle T, \mathbb{Z}, T \rangle$  dove  $\mathbb{Z}$  genera tutti i numeri interi.

- (a) Scrivere per ricorsione strutturale una funzione *sum* che, dato un albero, calcola la somma di tutti gli interi contenuti nell'albero.
- (b) Scrivere per ricorsione strutturale una funzione *prune* che, dato un albero, restituisca una copia dell'albero in cui tutti gli interi negativi sono stati rimpiazzati da 0.
- (c) Dimostrare per induzione strutturale che  $\forall T. \text{sum}(\text{prune}(T)) \geq 0$ . Preferire una prova intuizionista a una classica ove possibile.

Nello svolgere l'esercizio potete utilizzare come date le operazioni di somma e confronto ( $\geq$ ) fra numeri interi, nonché il lemma che dice che la somma di numeri interi non negativi è non negativa.

(a) Prima arte di ricorsione strutturale:

$$\begin{aligned} \text{sum}(\emptyset) &= 0 \\ \text{sum}(\langle T_1, x, T_2 \rangle) &= \text{sum}(T_1) + x + \text{sum}(T_2) \end{aligned}$$

(b) Seconda arte di ricorsione strutturale:

$$\begin{aligned} \text{prune}(\emptyset) &= \emptyset \\ \text{prune}(\langle T_1, x, T_2 \rangle) &= \langle \text{prune}(T_1), \text{if } (x \geq 0) \text{ then } x \text{ else } 0, \text{prune}(T_2) \rangle \end{aligned}$$

(c) Parte di induzione strutturale:

Teorema:  $\forall T. \text{sum}(\text{prune}(T)) \geq 0$ .

Dimostrazione: procediamo per induzione strutturale su  $T$  per dimostrare  $\text{sum}(\text{prune}(T)) \geq 0$

- Caso  $\emptyset$ : dobbiamo dimostrare  $\text{sum}(\text{prune}(\emptyset)) \geq 0$  o, equivalentemente,  $0 \geq 0$ . Ovvio per la proprietà riflessiva del  $\geq$ .
- Caso  $\langle T_1, x, T_2 \rangle$ : per ipotesi induttiva  $\text{sum}(\text{prune}(T_1)) \geq 0$  (II1) e  $\text{sum}(\text{prune}(T_2)) \geq 0$  (II2). Dobbiamo dimostrare  $\text{sum}(\text{prune}(\langle T_1, x, T_2 \rangle)) \geq 0$  o, equivalentemente,  $\text{sum}(\text{prune}(T_1)) + (\text{if } (x \geq 0) \text{ then } x \text{ else } 0) + \text{sum}(\text{prune}(T_2)) \geq 0$ . Dimostriamo che  $\text{if } (x \geq 0) \text{ then } x \text{ else } 0 \geq 0$  per casi su  $x \geq 0$ :
  - Caso  $x \geq 0 = \text{true}$ :  $\text{if } (\text{true}) \text{ then } x \text{ else } 0 = x \geq 0$  in quando  $x \geq 0$
  - Caso  $x \geq 0 = \text{false}$ :  $\text{if } (\text{false}) \text{ then } x \text{ else } 0 = 0 \geq 0$  per la proprietà riflessiva del  $\geq$

La tesi consegue dalle ipotesi induttive II1 e II2, da quanto appena dimostrato e dalla proprietà dei numeri interi che dice che la somma di numeri non negativi è non negativa.

Qed.

Cognome \_\_\_\_\_ Nome \_\_\_\_\_

Matricola \_\_\_\_\_ Numero di CFU \_\_\_\_\_ Fila 1

**Università degli Studi di Bologna, Corso di Laurea in Informatica  
Esame di LOGICA PER L'INFORMATICA (9 CFU), 16/02/2024**

*Utilizzare i riquadri bianchi per le risposte. Se strettamente necessario, si può allegare un foglio protocollo in coda con ulteriore testo, indicando in alto nome, cognome, fila e matricola.*

A5 (3 punti). Per ciascuno dei seguenti enunciati, indica se é vero o falso. Se falso, scrivi un controesempio.

	Linguaggio	V	F (scrivi controesempio)
(a)	$(\mathbb{R}, \times, 0)$ forma un monoide.		No, 0 non é l'elemento neutro per $\times$ .
(b)	<p>Considera il monoide <math>(\mathbb{L}, ++, [])</math> dove <math>\mathbb{L}</math> é l'insieme di liste di numeri naturali, <math>++</math> é la concatenazione di liste, e <math>[]</math> é la lista vuota.</p> <p>La funzione <math>f: \mathbb{L} \rightarrow \mathbb{L}</math> definita come <math>f(l) = 0 :: l</math> é un morfismo di monoidi da <math>(\mathbb{L}, ++, [])</math> a <math>(\mathbb{L}, ++, [])</math>.</p>		$f([1, 2] ++ [1]) = [0, 1, 2, 1]$ ma $f([1, 2]) ++ f([1]) = [0, 1, 2, 0, 1]$ .
(c)	$(\mathcal{P}(X), \cup, \emptyset, \cap)$ , dove $\mathcal{P}(X)$ é l'insieme dei sottoinsiemi di un dato insieme $X$ , é un semianello.	Si	

A6 (3 punti). Sia  $f: \mathbb{A} \rightarrow \mathbb{B}$  un morfismo tra strutture algebriche  $\mathbb{A}$  e  $\mathbb{B}$  dello stesso tipo.

- (a) Definisci cosa si intende per immagine di  $f$ , notazione  $Imm(f)$ .
- (b) Definisci cosa si intende per insieme quoziente di  $\mathbb{A}$  rispetto ad  $f$ , notazione  $\mathbb{A}/\sim_f$ .
- (c) Enuncia il teorema fondamentale dei morfismi per  $f: \mathbb{A} \rightarrow \mathbb{B}$ .

- (a) Sia  $B$  il sostegno di  $\mathbb{B}$ . L'immagine di  $f$  è il suo sottoinsieme  $\{b \in B \mid \exists a. f(a) = b\}$ .
- (b) Sia  $A$  il sostegno di  $\mathbb{A}$ . L'insieme quoziente ha come elementi le classi di equivalenza della relazione  $R$  su  $A$  definita da:  $(a, b) \in R$  se e solo se  $f(a) = f(b)$ .
- (c)  $\mathbb{A}/\sim_f$  è il sostegno di una struttura algebrica dello stesso tipo di  $\mathbb{A}$ , e tale struttura è isomorfa all'immagine di  $f$ .

A7 (4 punti). Sia  $(X, \circ, e, {}^{-1})$  un gruppo abeliano, con operazione binaria  $\circ$ , elemento neutro  $e$ , ed operazione unaria  ${}^{-1}$ . Sia  $Y \subseteq X$  un sottoinsieme non vuoto di  $X$ . Assumiamo che:

- (I)  $X$  è un insieme finito, diciamo di  $k$  elementi, che indichiamo come  $\{a_1, a_2, \dots, a_k\}$ .
- (II)  $Y$  è chiuso sotto l'operazione  $\circ$ , cioè  $x \in Y$  e  $y \in Y$  implica  $x \circ y \in Y$ .

Dimostra le seguenti proposizioni (A), (B), (C), e (D):

- (A) Dimostra che, dati  $a \in X$  e  $b \in X$ ,  $a \circ b = a$  implica  $b = e$ .
- (B) Definiamo per ricorsione sui numeri naturali:

$$\begin{aligned} a^0 &= e \\ a^{n+1} &= a \circ a^n \end{aligned}$$

Per cui, ad esempio,  $a^3 = a \circ (a \circ a)$ . Dimostra che esiste  $j > 0$  tale che  $a^j = e$ .

- (C) Dimostra che, se  $a \in Y$ , allora  $a^{-1} \in Y$ .
- (D) Dimostra che  $(Y, \circ, e, {}^{-1})$  forma un sottogruppo di  $(X, \circ, e, {}^{-1})$ .

Nota che nel dimostrare ciascuna proposizione puoi assumere le precedenti come date.

Cognome \_\_\_\_\_ Nome \_\_\_\_\_

Matricola \_\_\_\_\_ Numero di CFU \_\_\_\_\_ Fila 1

**Università degli Studi di Bologna, Corso di Laurea in Informatica  
Esame di LOGICA PER L'INFORMATICA (9 CFU), 16/02/2024**

*Utilizzare i riquadri bianchi per le risposte. Se strettamente necessario, si può allegare un foglio protocollo in coda con ulteriore testo, indicando in alto nome, cognome, fila e matricola.*

- Dimostriamo che, se  $a = a \circ b$ , allora  $b$  deve essere  $e$ . Abbiamo che  $e = a^{-1} \circ a = a^{-1} \circ (a \circ b) = (a^{-1} \circ a) \circ b = e \circ b = b$ .
- Dal momento che  $X$  é finito, esiste  $j > 0$  per cui  $a^{n+j} = a^n$ . Per definizione,  $a^{n+j} = a^n \circ a^j$ . Perciò  $a^n \circ a^j = a^n$ , che implica  $a^j = e$  per la proposizione precedente.
- Supponi  $a \in Y$ . Per la proposizione precedente, abbiamo  $m$  tale che  $a^{m+1} = e$ . Perciò  $a \circ a^m = a^{m+1} = e$ . Quindi  $a^m$  é l'elemento inverso di  $a$ . (Nota che non dobbiamo anche dimostrare  $a^m \circ a = e$  in quanto il gruppo é abeliano.) Per definizione di  $a^n$ , dal momento che per assunzione  $Y$  é chiuso rispetto all'operazione  $\circ$ , abbiamo che  $a^n \in Y$  per ogni  $n > 0$ . Inoltre, per  $n = 0$ , abbiamo anche che  $a^n = e$  é in  $Y$  per la proposizione precedente. Perciò in particolare l'inverso  $a^m$  di  $a$  é in  $Y$ .
- Abbiamo già dimostrato che  $Y$  é chiuso rispetto all'operazione  $^{-1}$ , e per assunzione é chiuso rispetto all'operazione  $\circ$ . Rimane da dimostrare che contiene l'elemento neutro  $e$ . Dato  $a \in Y$ , abbiamo che  $a^{-1} \in Y$ , e anche  $a \circ a^{-1} \in Y$ . Dato che  $a \circ a^{-1} = e$ , questo conclude la dimostrazione. Alternativamente, possiamo utilizzare Proposizione (B).