Università degli Studi di Bologna

Corso di Laurea in Informatica
Esame scritto di LOGICA PER L'INFORMATICA (9/6 CFU)
26/05/2022

Scrivere **nome, cognome, numero di CFU e numero di matricola in alto a destra** in tutti i fogli protocollo. Gli esercizi con un doppio punteggio riportano prima il punteggio nel caso dell'esame da 9 CFU e poi quello nel caso di esame da 6 CFU. Fare attenzione all'esercizio 10 che è diverso a seconda del numero di CFU.

1 (1 punto). Dare la sintassi per le formule della logica proposizionale.

2 (4 punti/6 punti). Considerare la seguente sintassi per alberi binari i cui nodi siano etichettati con numeri interi: $T ::= \emptyset \mid \langle T, \mathbb{Z}, T\rangle$ dove il non terminale $\mathbb{Z}$ genera tutti i numeri interi.

Scrivere, per ricorsione strutturale, una funzione $f(T_1) = \langle T_2, p\rangle$ dove T_1 e T_2 siano alberi in accordo alla precedente grammatica, T_2 sia un sotto-albero di T_1, p sia la somma di tutti i numeri contenuti nei nodi di T_2 e non vi sia nessun altro sotto-albero di T_1 la cui somma dei numeri contenuti sia maggiore di p. L'output della funzione f è una coppia generata dalla grammatica $C ::= \langle T, \mathbb{Z}\rangle$.

Es:

$f(\emptyset) = \langle \emptyset, 0\rangle$
$f(\langle\langle\langle \emptyset, -2, \emptyset\rangle, 3, \langle \emptyset, 2, \emptyset\rangle\rangle, 2, \langle \emptyset, -3, \emptyset\rangle\rangle) = \langle\langle\langle \emptyset, -2, \emptyset\rangle, 3, \langle \emptyset, 2, \emptyset\rangle\rangle, 3\rangle$
$f(\langle\langle\langle \emptyset, -2, \emptyset\rangle, 1, \langle \emptyset, 2, \emptyset\rangle\rangle, -1, \langle \emptyset, 1, \emptyset\rangle\rangle) = \langle\langle \emptyset, 2, \emptyset\rangle, 2\rangle$

3 (5 punti). Dimostrare in teoria assiomatica degli insiemi che

$$\forall A \forall A' \forall B \forall B' (A \subseteq A' \land B \subseteq B' \Rightarrow A \setminus B' \subseteq A' \setminus B)$$

dove $X \setminus Y := \{Z \in \mathcal{U} \mid Z \in X \land Z \notin Y\}$ e $\mathcal{U}$ è un insieme (detto universo) fissato.

Prima della dimostrazione riportare l'enunciato di tutti gli assiomi di teoria degli insiemi che usate nella dimostrazione.

4 (1 punto). Enunciare il teorema di correttezza per la logica proposizionale classica.

5 (1 punto). Dare la definizione di funzione in teoria degli insiemi.

6 (1 punto). Quali delle seguenti coppie di formule logiche non sono logicamente equivalenti? Per quelle che non lo sono fornire un'interpretazione (A, I, ξ) che ne renda una vera e una falsa.

(a) $\forall x \exists y.P(x, y)$ vs $\exists y \forall x.P(x, y)$

(b) $\forall x.\forall y.P(x, y)$ vs $\forall x.P(x, x)$

7 (5 punti). Considerare gli alberi e una funzione f, soluzione dell'esercizio 2, che, nel caso di soluzioni multiple possibili. Considerare il predicato $Pos(T)$ che dice che tutti i numeri nei nodi di T sono strettamente positivi, definito per ricorsione strutturale come segue:

$$Pos(\emptyset) = \top, \quad Pos(\langle T_1, z, T_2 \rangle) = Pos(T_1) \wedge z > 0 \wedge Pos(T_2)$$

Dimostrare, per induzione strutturale su T, che

$$\forall T(Pos(T) \Rightarrow f(T) = \langle T, sum(T) \rangle)$$

.

8 (6 punti). Si consideri il seguente ragionamento:

Gli USA attaccheranno o l'Ucraina non vincerà la guerra. Se Putin verrà deposto allora l'Ucraina e l'Unione Europea vinceranno la guerra. Se l'Unione Europea vincerà la guerra e gli USA attaccheranno allora la Cina interverrà. Se l'Ucraina vincerà la guerra o la Cina interverrà, allora gli USA non attaccheranno. Quindi l'Ucraina non vincerà la guerra e se Putin verrà deposto allora la Cina interverrà.

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

9 (1 punto/2 punti). Effettuare la seguente sostituzione minimizzando il numero di cambi di nome alle variabili.

$$(\forall x \exists y \Sigma_{i=x-a}^{y+b} a * i)[(a * x - i * z)/b]$$

10 (2 punti). **PER CHI SOSTIENE L'ESAME DA 6 CFU**
Dimostrare il seguente teorema usando la deduzione naturale al prim'ordine,

preferendo una prova intuizionista a una classica ove possibile. Considerare la somma associativa a destra.

$$(\forall x.\exists y.x \leq k + y + (-k)) \Rightarrow \forall y.\exists x.k + y \leq k + x$$

10 (5 punti). **PER CHI SOSTIENE L'ESAME DA 9 CFU**

Si consideri il seguente codice dove `^` concatena due stringhe, `""` è la stringa vuota e `to_string(x)` restituisce la rappresentazione di x come stringa.

Il senso della funzione $len(l)$ è quello di calcolare la lunghezza della stringa l assieme a una stringa di logging (usata per il debugging) che mostra l'elenco di tutti i numeri visitati separati da spazio.

```
fst(<a,b>) = a
snd(<a,b>) = b

len([]) = < 0, "" >
len(x:xs) = < 1 + fst(len(xs)), to_string(x) ^ " " ^ snd(len(xs)) >
```

- Cosa restituisce la chiamata $len(1 : 2 : 3 : 4 : [])$?
- Come programmatori vi viene richiesto di scrivere una variante della funzione `len` che, invece di restituire come secondo parametro il log dei numeri processati, restituisca il massimo numero contenuto nella lista. Risolvete il problema generalizzando la funzione $len()$ usando una type class e mostrare tre istanze differenti che corrispondano alla `len` originale, a quella che calcola anche il massimo della lista in input e a quella che calcola anche la somma degli elementi della lista (ovvero $len(1 : 2 : 3 : 4 : []) =< 4, "1234" >$, $len(1 : 2 : 3 : 4 : []) =< 4, 4 >$ e $len(1 : 2 : 3 : 4 : []) =< 4, 10 >$)
- Definire l'istanza "prodotto cartesiano" della type class che possa essere poi istanzata per permettere di calcolare al tempo stesso la lunghezza della lista in input e il log dei numeri incontrati.