Prova Scritta di Linguaggi di programmazione
Corso di Linguaggi di programmazione I
06/07/2004

Prolog

Scrivere un predicato `almostBalanced/1` in SICSTUS PROLOG per determinare se un albero binario ha la seguente proprietà: per ogni nodo le altezze dei figli destro e sinistro differiscono al massimo di 1. L'altezza è la lunghezza del massimo cammino. Si ricordi che si possono usare i funzionali `abs` e `max`.
Ad esempio `almostBalanced(tree(4,void,tree(3,void,void)))` è vero.
Si scriva il programma con variabili anonime ove possibile.

Unificazione

Si dica se esiste un unificatore più generale di `f(g(X),X,h(X))` e `f(T,g(Z),h(T))` e in caso affermativo lo si calcoli.

Semantica

Si consideri il seguente frammento di programma espresso in un linguaggio imperativo C-like con scoping statico:

```
{   int x:=7;
    proc Q(value int y; ref int w)
    {
            x := 3*y;
            w := w+1;
            y := w+2*y;
    }
Q(x, x);
}
```

si fornisca la semantica denotazionale del frammento, mostrando l'ambiente e la memoria nel momento in cui termina la valutazione della chiamata di procedura Q.

NOME E COGNOME:	Numero di Matricola:

Prolog

```
almostBalanced(void, 0).
almostBalanced(tree(_,L,R), D) :-
  almostBalanced(L, DL),
  almostBalanced(R, DR),
  Diff is abs(DL-DR),
  Diff =< 1,
  D is 1+max(DL,DR).
```

Unificazione

Non esiste unificatore.

Linguaggi di Programmazione I

Esame del 6 Luglio 2004

cognome e nome

Macchina astratta Pascal

Con riferimento al seguente programma in Pascal, si rappresenti in questo foglio il P-code relativo al corpo della procedura *gcd*, cioè il P-code corrispondente alla sezione evidenziata dal riquadro.

```
program  esercizio;     . . .     . . .     . . .
    procedure  gcd( x, y: integer;   var z: integer );
        var  r: integer;
    begin
        r := x mod y;
        if   r = 0   then   z := y
                     else   gcd( y, r,  z )
    end;
begin     . . .     . . .     . . .     end.
```