

CORSO DI PARADIGMI DI PROGRAMMAZIONE
PROVA SCRITTA DEL 20 SETTEMBRE 2005.

Tempo a disposizione: ore 2.

1. Con la notazione \mathcal{C}_{L_1, L_2}^L indichiamo un compilatore da L_1 a L_2 scritto in L . Con $\mathcal{I}_{L_1}^L$ indichiamo un interprete scritto in L per il linguaggio L_1 ; se P è un programma in L_1 e x un suo dato, $\mathcal{I}_{L_1}^L(P, x)$ indica l'applicazione dell'interprete a P e x . Si dica cosa produce la valutazione di $\mathcal{I}_{L_1}^L(\mathcal{C}_{L, L_1}^{L_1}, \mathcal{I}_{L_1}^L)$ e se un tale risultato può effettivamente esistere.
2. Si fornisca un esempio di grammatica ambigua (specificando perché si ha l'ambiguità).
3. Si dica cosa viene stampato dal seguente frammento di codice scritto in uno pseudo-linguaggio che usa scope dinamico e passaggio di parametri per riferimento. La primitiva `write(x,y,z)` permette di stampare i valori dei tre argomenti.

```
{int x = 2;
 int y = 6;
 int z = 10;
void pluto(reference int v, reference int w){
    int x = 1;
    w = v;
    v = y+w+z;
    z = 11;
}
 { int x = 20;
  int y = 60;
  int z = 100;
  pluto(x, y);
  write(x,y,z);
}
write(x,y,z) }
```

4. Si dica cosa viene stampato dal seguente frammento in un linguaggio con eccezioni:

```
{
void f() throws X {
    throw new X();
}

void g (int sw) throws X {
    if (sw == 0) {f();}
    try {f();} catch (X e) {write("in_g");}
}
...
try {g(1);}
    catch (X e) {write("in_main");}
}
```

5. Si consideri la seguente definizione in uno pseudolinguaggio che ammette ricorsione e dove il passaggio dei parametri avviene per valore:

```
int f(int n, int m){
    if (n==0) return m;
    return f(n-1,m+n);
}
```

Il codice è tradotto con il migliore compilatore ottimizzante presente sul mercato. Quanti RdA per `f` devono essere messi sulla pila di sistema per il calcolo di `f(5,0)`?

6. In un linguaggio che permette overloading, si possono scrivere le seguenti quattro espressioni

```
1+2
1.0+2.0
1+2.0
1.0+2
```

Com'è possibile che tutte e quattro le espressioni siano corrette, sapendo che `+` ha due soli significati sovraccaricati?

7. Si consideri la seguente definizione di tipo record:

```
type S = struct{
  int x;
  int y;
};
```

Si supponga che un `int` sia memorizzato su 2 byte, su un'architettura a 16 bit con allineamento alla parola. In un blocco viene dichiarato un vettore:

```
s A[20];
```

Indicando con `PRDA` il puntatore all'RdA di tale blocco, e con `ofst` l'offset tra il valore di `PRDA` e l'indirizzo iniziale di memorizzazione di `A`, si dia l'espressione per il calcolo dell'indirizzo dell'elemento `A[3].y`.

8. **Solo per il corso AL** Si scriva un programma logico tale che la valutazione del goal `p(X)` non termina se si usa la regola di selezione da sinistra a destra, mentre termina se si usa la regola di selezione da destra a sinistra.
9. **Solo per il corso MZ** Si descriva brevemente la differenza tra valutazione per nome e valutazione *lazy* nel caso di un linguaggio di programmazione funzionale.