

# Esame di LP1

Prof Piero Bonatti

9 Settembre 2014

## Domande generali – Max 6 punti

**Esercizio 1: [2 punti]** Barrare tutte le frasi vere.

- a)  Il sistema di tipi del C adotta solo la name equivalence.
- b)  Il sistema di tipi del C adotta solo la structural equivalence.
- c)  In Prolog, la ricerca del massimo in una lista può essere fatta in forma iterativa (non ricorsiva).
- d)  In un linguaggio funzionale puro, un identificatore  $x$  in una espressione rappresenta  $env(x)$ .
- e)  In C, l'identificatore  $x$  in  $y=x$ ; rappresenta  $env(x)$ .
- f)  In C, l'espressione  $\&x$  in  $y=\&x$ ; rappresenta  $env(x)$ .

**Esercizio 2: [2 punti]** Disegnare i data object generati dal seguente codice C:

```
int x, *y;  
y = &x;  
*y = 0;
```

**Esercizio 3: [2 punti]** Il diagramma qui sotto rappresenta l'implementazione di Java su piattaforma Linux. Riempire ognuno dei 7 rettangoli con la lettera corrispondente all'entità che rappresenta.

A – Compilatore Java

B – Java virtual machine

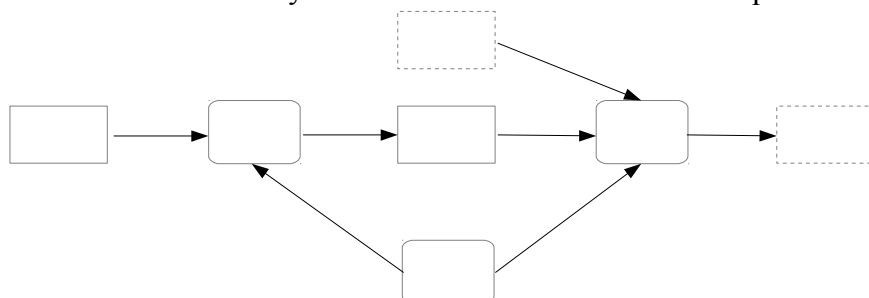
C – Piattaforma Linux

D – Sorgente Java

E – Bytecode

F – Input

G – Output



## Esercizio sul passaggio di parametri – Max 11 punti

Dire qual è l'output del seguente programma nei casi elencati qui sotto:

1. Scoping dinamico, [MODE] = IN OUT per riferimento
2. Scoping dinamico, [MODE] = IN per riferimento
3. Scoping statico, [MODE] = IN OUT per copia
4. Scoping statico, [MODE] = IN OUT per riferimento

Mostrare gli stack di attivazione (pena la perdita di punti), tranne nei casi di errore, nei quali bisogna invece indicare l'istruzione che causa l'errore.

```
program p1
int a; int b; int c; int d;
  procedure p2([MODE] int a)
    int b; int d;
    BEGIN
    b=a+4;
    d=3;
    a=2;
    c=a*3;
    p3(d);
    write(a,b,c,d);
    END

  procedure p3([IN x copia] int a)
    int b;
    procedure p4([IN OUT x copia] int d,[IN OUT x rif] int c)
      int b;
      BEGIN
      b=a-2;
      d=c*2;
      c=c*4;
      a=1;
      write(a,b,c,d);
      END

    BEGIN
    b=d-4;
    a=4;
    c=a;
    d=3;
    p4(d, a);
    write(a,b,c,d);
    END

BEGIN
a=4;
b=0;
c=2;
d=4;
p2(c);
write(a,b,c,d);
END
```

## UML – Max 15 punti

Progettare un sistema bancario in cui i clienti sono caratterizzati da nome e codice fiscale. Ogni cliente è associato a uno o più conti correnti; questi ultimi sono caratterizzati da un saldo e una lista concatenata di movimenti (costituiti da codice e ammontare dell'operazione) ordinata cronologicamente. I conti correnti possono essere cointestati. Ai conti correnti si possono aggiungere nuovi movimenti (specificando codice e ammontare). Inoltre con i conti correnti si possono effettuare bonifici, specificando ammontare e codice IBAN del destinatario, utilizzando una componente interbancaria realizzata da altri di cui si sa solo che supporta il metodo *bonifico* con i suddetti input.

**Esercizio 1:** Disegnare un diagramma delle classi per queste specifiche.  
**[max 8 punti]**

**Esercizio 2:** Disegnare un diagramma delle sequenze per lo scenario seguente, mostrando tutte le interazioni tra gli oggetti coinvolti.

Un cliente aggiunge al conto corrente CC1 un nuovo movimento con codice 11 e ammontare 100€. Questo viene appeso alla lista di movimenti di CC1 mediante un algoritmo ricorsivo. Al momento dell'inserimento la lista contiene già 2 movimenti.

**[max 7 punti]**