

Tempo a disposizione: ore 2.

1. Si dica cosa stampa il seguente frammento in uno pseudolinguaggio con passaggio per nome e scope statico

```
int x = 5;
void foo(name int y){
    x = x-1;
    y = y+10;
    x = x+y;
    write(x);
}
{int x = 12;
  foo(x);
  write(x);
}
```

2. Il linguaggio imperativo Ric è costituito dagli usuali comandi (assegnamenti, controllo di sequenza ecc.), non permette comandi di allocazione (e deallocazione) esplicita della memoria, ammette funzioni, ma, nel caso di funzioni ricorsive, queste devono essere ricorsive in coda. Si dica, motivando la risposta, qual è la più semplice forma di gestione della memoria utilizzabile nell'implementazione di Ric.
3. L'esecuzione del seguente frammento di codice su una certa implementazione risulta nella stampa del valore 9.

```
int V[10];
int x = 4;
for (int i=0, i<10, i++) V[i]=i;
V[x] = V[x++] + V[x++];
write (V[x]);
```

Si fornisca una possibile spiegazione.

4. Si consideri il seguente frammento in uno pseudolinguaggio con scope statico e parametri di ordine superiore:

```
{void foo (int f(), int n){
  int m = 20;
  int fie(){
    write(n,m);
  }
  if (n==0) f();
  else {m = 21;
        foo(fie,0);
      }
}
int g(){
  write(10);
}
foo(g,1);
}
```

Si dica cosa stampa il frammento con (i) shallow binding; (ii) deep binding.

5. Sono date le seguenti dichiarazioni in un linguaggio ad oggetti con classi:

```
class A {
  int x=10;
  void f(){write(x);}
  void g(){f();}
}
class B extending A {
  int x=20;
  void f(){write(x);}
}
```

Si dica cosa stampa il seguente frammento, nel contesto di tali dichiarazioni:

```
A a = new A();
a.g();
a.f();
B b = new B();
a = b;
a.g();
a.f();
write (a.x);
write (b.x);
b.g();
```

6. Si discuta brevemente la differenza esistente fra polimorfismo universale parametrico e polimorfismo universale di sottotipo.
7. Si considerino le seguenti definizioni di classe in Java:

```
class A{
    int x;
    int f (int y){return y+1;}
}
class B extends A{
    int y;
    void g (int z){...}
}
class C extends B{
    int f (int y){return y+2;}
}
```

Si supponga che la gerarchia delle classi sia implementata mediante vtable. Qual è la struttura della vtable di C?

8. Usando uno pseudolinguaggio che usi puntatori si fornisca un frammento di codice che generi un “dangling reference”. Si faccia quindi vedere come con la tecnica dei “locks and keys” non si ha più tale problema.