

Esercizio: ancora su overloading e gerarchia

Considerare il seguente codice java:

```
package it.uniud.poo.abstractions;
public class Overloading {
    static int i = 0;
    class D {
        void m (D d){ System.out.format("%3d D.m(D %s)\n",i++,d); }
        void m (C c1){ System.out.format("%3d D.m(C %s)\n",i++,c1); }
        void m (B b1){ System.out.format("%3d D.m(B %s)\n",i++,b1); }
        void k (D d1, D d2){System.out.format("%3d D.k(D D)\n",i++); };
        void k (C c1, D d2){System.out.format("%3d D.k(C D)\n",i++);};
        void k (D d1, C c1){System.out.format("%3d D.k(D C)\n",i++);};
    }

    class C extends D {
        void m (D d1){ System.out.format("%3d C.m(D %s)\n",i++,d1); }
        void m (C c1){ System.out.format("%3d C.m(C %s)\n",i++,c1); }
        void m (B b1){ System.out.format("%3d C.m(B %s)\n",i++,b1); }
    }

    class B extends D {
        void m (D d1){ System.out.format("%3d B.m(D %s)\n",i++,d1); }
        void m (B b1){ System.out.format("%3d B.m(B %s)\n",i++,b1); }
        void m (C c1){ System.out.format("%3d B.m(C %s)\n",i++,c1); }
    }

    static Overloading o = new Overloading();

    public static void main (String a[]){
        D aD = o.new D();// o.new() needed to create an instance of the inner class D
        C aC = o.new C();
        D anotherD = o.new C();
        B aB = o.new B();

        aD.m(aD);
        aD.m(aC);
        aD.m(anotherD);
        aD.m(aB); System.out.println();

        aC.m(aD);
        aC.m(aC);
        aC.m(anotherD);
        aC.m(aB); System.out.println();

        anotherD.m(aD);
        anotherD.m(aC);
    }
}
```

```

        anotherD.m(anotherD);
        anotherD.m(aB); System.out.println();

        aB.m(aD);
        aB.m(aC);
        aB.m(anotherD);
        aB.m(aB);

        aD.k(aD, aD);
        aD.k(aD, aC);
        aD.k(aC, aC);
    }
}

```

Quali delle invocazioni dentro il body di **Main** sono corrette, e quelle corrette quale definizione invocherebbero?

1. Deciderlo in via analitica
2. Implementarlo e provarlo.