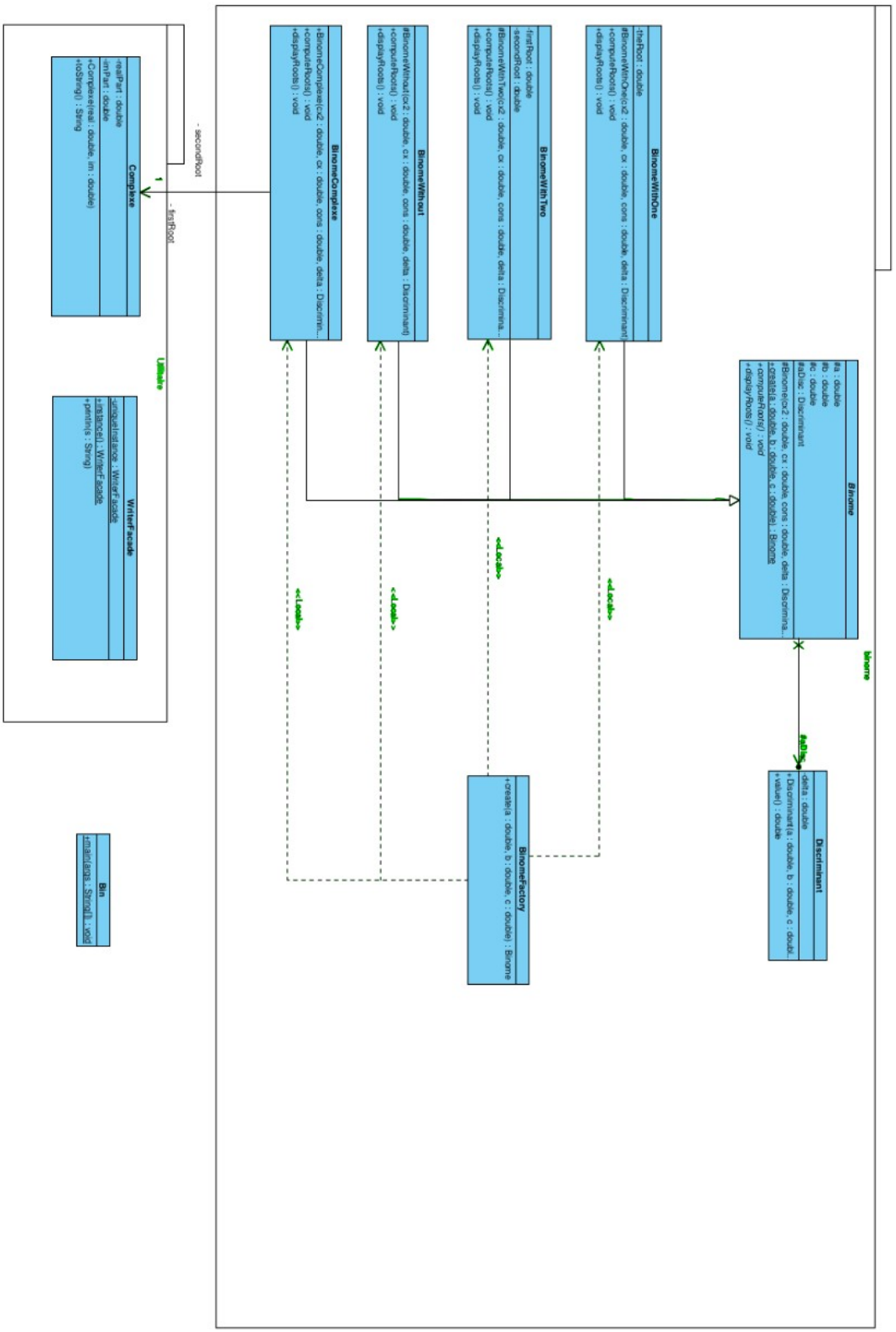
M3105 : TP5
Antoine Gicquel

Le diagramme :



Le code :

```
package binome;

public class BinomeFactory {

  public Binome create(double a, double b, double c) {
      Discriminant d = new Discriminant(a, b, c);
      double delta = d.value();
      Binome aBin;

      if ( delta == 0.0 ) {
          aBin = new BinomeWithOne (a, b, c, d);
      } else if ( delta > 0.0 ) {
          aBin = new BinomeWithTwo (a, b, c, d);
      } else {
          aBin = new BinomeComplexe (a, b, c, d);
      }
      return aBin;
  }
}


import binome.*;
public class Bin {
    public static void main(String args[]) {

        BinomeFactory b = new BinomeFactory();
        Binome bin;

        bin = b.create (1.0, 0.0, 1.0);
        bin.computeRoots();
        bin.displayRoots();
        bin = b.create (1.0, 0.0, -1.0);
        bin.computeRoots();
        bin.displayRoots();
        bin = b.create (1.0, 2.0, 1.0);
        bin.computeRoots();
        bin.displayRoots();
    }
}


package Utilitaires;

public class WriterFacade {
  private static WriterFacade uniqueInstance;

  public static WriterFacade instance() {
    if(uniqueInstance == null) {
      uniqueInstance = new WriterFacade();
    }

    return uniqueInstance;
  }

  public void println(String s) { System.out.println(s); }
}
```

```
package binome;
import Utilitaires.*;

class BinomeWithOne extends Binome {

    [...]

    public void displayRoots() {
        WriterFacade.instance().println("Une racine double : " + theRoot);
    }

}


package binome;
import Utilitaires.*;

class BinomeWithTwo extends Binome {

    [...]

    public void displayRoots() {
        WriterFacade.instance().println("Deux racines distinctes : \n\tx1 = " +
                            firstRoot);
        WriterFacade.instance().println("\tx2 = " + secondRoot);
    }

}


package binome;
import Utilitaires.*;

class BinomeComplexe extends Binome {

  [...]

  public void displayRoots() {
    WriterFacade.instance().println("Deux racines distinctes : \n\tx1 = " +
firstRoot.getReal() + " + " + firstRoot.getIm()  + "i");
    WriterFacade.instance().println("                           \n\tx2 = " +
secondRoot.getReal() + " + " + secondRoot.getIm()  + "i");
  }

}
```