Question 1 ......................................................................................... (6 points)
Circle and describe all syntax errors in the following program fragment:

float value_1;
float value_2 = 2.0;
if (value_1=value_2) {
    System.out.println("equal");
} else {
    System.out.println("not equal")
}

Question 2 ......................................................................................... (8 points)
In the Java statement

public static final float SQRT_TWO = (float)Math.sqrt(2.0);

what is the purpose of

• the keyword static?
• the keyword final?
• the operator (float)?

Which of these if omitted would cause a syntax (compile) error?

Question 3 ......................................................................................... (6 points)
Consider the following method:

public static void mystery(int n) {
    int num = 0;
    while (n>0) {
        int k = n/10*10;
        for (int i=k; i<n; ++i)
            ++num;
        n = k/10;
    }
    System.out.println(num);
}

What is printed by mystery(1)? By mystery(123)? By mystery(12345)?
Question 4 .......................................................... (15 points)

(a) [5 points] Returns true, if \( j \) is evenly divisible by 2, \( k \) is evenly divisible by 3, and \( j \) is evenly divisible by \( k \); returns false, otherwise.

\[
\text{public static boolean divisible(int } j, \text{ int } k) \{ \\
\}
\]

(b) [5 points] Returns the smallest of four specified \textit{floats}.

\[
\text{public static float smallest(float } a, \text{ float } b, \text{ float } c, \text{ float } d) \{ \\
\}
\]

(c) [5 points] Draws a grid of \( n \) horizontal lines and \( n \) vertical lines that are evenly spaced and fill a panel with specified width \( w \) and height \( h \).

\[
\text{public void drawGrid(Graphics } g, \text{ int } n, \text{ int } w, \text{ int } h) \{ \\
\}
\]
Implement all methods for the following classes:

```java
/**
 * A waypoint (geographic location) has a name, latitude and longitude.
 */
public class Waypoint {

    /** Constructs a waypoint with specified name and zero lat and long. */
    public Waypoint(String name) {
    }

    /** Sets the location for this waypoint. */
    public void setLocation(double latitude, double longitude) {
    }

    /** Gets the name for this waypoint. */
    public String getName() {
    }

    /** Gets the latitude for this waypoint. */
    public double getLatitude() {
    }

    /** Gets the longitude for this waypoint. */
    public double getLongitude() {
    }

    /**
     * Determines whether this waypoint equals the specified waypoint.
     * Two waypoints are equal if they have the same name and location.
     */
    public boolean equals(Waypoint wp) {
    }
}
```
/**
 * Returns a clone of this waypoint with the specified name.
 * The clone has the same lat and long as this waypoint.
 */
public Waypoint clone(String name) {

}

/**
 * Using the methods defined in the class Waypoint,
 * (1) constructs a waypoint for a location home named "Home",
 * (2) sets the location of the home waypoint,
 * (3) creates a clone of the home waypoint with name "Mines",
 * (4) gets and prints the location of the new waypoint.
 */
public class WaypointDemo {
    public static void main(String[] args) {

    }