As usual you will work in pairs and only one of you needs to submit the files. Here we give you the exact main program that you must use and your task is to create the methods needed to complete the program.

1. For Number 1 just submit just the Java code file containing the program.

2. For Number 2 submit the Java code file containing the program and a Word, LibreOffice, or text file containing at least 3 runs of the program with different initial heights and velocities.

1   Exercises

1. The first program is to construct a Fahrenheit to Celsius conversion chart that will print out a chart of Fahrenheit to Celsius conversions. The output should look like the following.

   ```
   Fahrenheit     Celsius
     -50.00       -45.56
     -45.00       -42.78
     -40.00       -40.00
     -35.00       -37.22

   <<< Removed for Handout >>>

    135.00        57.22
    140.00        60.00
    145.00        62.78
    150.00        65.56
   ```

   The main program that you must use is,

   ```java
   public static void main(String[] args) {
       System.out.println("Fahrenheit     Celsius");

       for (int i = -50; i <= 150; i += 5) {
           System.out.printf("%6d    %10.2f\n", i, Celsius(i));
       }
   }
   ```

   You need to write the method `Celsius` that takes in a single parameter, a double data type, of the Fahrenheit temperature and return the corresponding Celsius temperature, also a double data type. The conversion formula for Fahrenheit to Celsius is

   $$C = \frac{5}{9}(F - 32)$$

   where C is the Celsius temperature and F is the Fahrenheit temperature.

2. Most computer games use physics engines to take care of many aspects of game play. The most work that a physics engine usually does in a game is collision detection. Determining when the user (or object) hits a wall, when the laser blast or bullet in a

---

FPS hits a target, etc. Collision detection is a fairly difficult problem to solve and is beyond the scope of this class, but another, more accessible, use of a physics engine in a game is to apply gravity to an object.

You have probably seen this in high school physics or possibly as an example in a mathematics class like Calculus. If not, we will discuss the equation fully. Say you take an object, like a tennis ball or small rock, and through it up into the air. The object will have a particular height off the ground at any time after you toss it until it eventually hits the ground. The object will increase in height, until gravity overcomes its initial velocity, and then it will start to fall to the ground. To find the height of the object at any time t in seconds after you toss the object is given by the equation,

$$d = \frac{1}{2}gt^2 + v_0 t + d_0$$

where d is the current height of the object, g is the acceleration of gravity, v_0 is the initial velocity and d_0 is the initial height. The acceleration of gravity on Earth in English units is 32.17405 feet/sec^2, so velocity is in feet/sec, distance is in feet and time is in seconds. Also note that we usually consider the positive direction as up. So a positive d_0 would indicate that the initial position of the object is above the ground and a positive v_0 would indicate that the initial velocity is in the upward direction. Along these lines, gravity would be pulling the object down and hence it should be negative.

A sample run of the program is below.

```
Input the Initial Height (in feet): 10
Input the Initial Velocity (in feet/sec.): 25

 Time Height
  0.0          10.000
  0.1          12.339
  0.2          14.357
  0.3          16.052
  0.4          17.426
  0.5          18.478
  0.6          19.209
  0.7          19.617
  0.8          19.704
  0.9          19.470
  1.0          18.913
  1.1          18.035
  1.2          16.835
  1.3          15.313
  1.4          13.469
  1.5          11.304
  1.6           8.817
  1.7           6.008
  1.8           2.878
  1.9       Hit the ground.
```

The main program that you must use is,

```java
public static void main(String[] args) {
    Scanner keyboard = new Scanner(System.in);
    System.out.print("Input the Initial Height (in feet): ");
```

```java
        Double d0 = keyboard.nextDouble();
        System.out.print("Input the Initial Velocity (in feet/sec.): ");
        Double v0 = keyboard.nextDouble();

        double d = d0;
        double t = 0;

        System.out.println();
        System.out.println(" Time Height");

        while (d > 0) {
            d = height(d0, v0, t);
            if (d > 0)
                System.out.printf("%5.1f %15.3f \n", t, d);
            else
                System.out.printf("%5.1f %20s \n", t, "Hit the ground.");
            t += 0.1;
        }
    }
}
```

You need to write the method `height` that takes in three parameters all double values. The first is the initial height, the second the initial velocity, and the third is the time. The return value is the current height of the object, also a double data type.