Name: _____

1.

Consider the code segment below.

```
int x = 10;
int y = 20;
System.out.print(y + x / y);
```

What is printed as a result of executing the code segment?

A) 1
B) 1.5
C) 3
D) 20
E) 20.5

2.

Consider the following code segment.

```
int num = 5;
num *= 2;
num %= 6;
```

What is the value of num after the code segment is executed?

A) 1
B) 2
C) 4
D) 6
E) 10

3.

Consider the following class declaration.

```
public class Thing
{
  private String color;

  public Thing()
  {
    color = "Blue";
  }

  public Thing(String setColor)
  {
    color = setColor;
  }
}
```

Which of the following code segments, when appearing in a class other than Thing, would create a reference of type Thing with a value of null ?

A) Thing someThing = new Thing("Green");
B) Thing someThing = new Thing("");
C) Thing someThing = new Thing();
D) Thing someThing;
E) Thing("Green");

4.

Consider the following code segment.

```
String s1 = "ABCDEFGHI";
String s2 = s1.substring(6, 7);
String s3 = new String("abcdefghi");
String s4 = s3.substring(4, 5);
String s5 = s3.substring(2, 3);
System.out.print(s2 + s4 + s5);
```

What, if anything, is printed when the code segment is executed?

A) Fdb
B) Fgdebc
C) Gec
D) GHefcd
E) There is no output due to a compilation error.

5.

Consider the following code segment.

```
int a = 1;
int b = 0;
int c = -1;
if ((b + 1) == a)
{
  b++;
  c += b;
}
if (c == a)
{
  a--;
  b = 4;
}
```

What are the values of a, b, and c after this code segment has been executed?

A) a = 0, b = 4, and c = 0
B) a = 0, b = 4, and c = 1
C) a = 1, b = 0, and c = -1
D) a = 1, b = 1, and c = 0
E) a = 1, b = 1, and c = 1

6.

Consider the following code segment, which is intended to store the sum of all multiples of 10 between 10 and 100, inclusive (10 + 20 + ... + 100), in the variable total.

```
int x = 100;
int total = 0;
while( /* missing code */ )
{
  total = total + x;
  x = x - 10;
}
```

Which of the following can be used as a replacement for /* missing code */ so that the code segment works as intended?

A) x < 100
B) x <= 100
C) x > 10
D) x >= 10
E) x != 10

7.

Consider the following definition of the class Student.

```
public class Student
{
  private int grade_level;
  private String name;
  private double GPA;
  public Student (int lvl, String nm, double gr)
  {
    grade_level = lvl;
    name = nm;
    GPA = gr;
  }
}
```

Which of the following object initializations will compile without error?

A) Student max = new Student ("Max", 10, 3.75);
B) Student max = new Student (3.75, "Max", 10);
C) Student max = new Student (3.75, 10, "Max");
D) Student max = new Student (10, “Max”, 3.75);
E) Student max = new Student (10, 3.75, “Max”);

8.

The Fibonacci numbers are a sequence of integers. The first two numbers are 1 and 1. Each subsequent number is equal to the sum of the previous two integers. For example, the first seven Fibonacci numbers are 1, 1, 2, 3, 5, 8, and 13.

The following code segment is intended to fill the fibs array with the first ten Fibonacci numbers. The code segment does not work as intended.

```
int[] fibs = new int[10];
fibs[0] = 1;
fibs[1] = 1;
for (int j = 1; j < fibs.length; j++)
{
  fibs[j] = fibs[j - 2] + fibs[j - 1];
}
```

Which of the following best identifies why the code segment does not work as intended?

A) In the for loop header, the initial value of j should be 0.
B) In the for loop header, the initial value of j should be 2.
C) The for loop condition should be j < fibs.length - 1.
D) The for loop condition should be j < fibs.length + 1.
E) The for loop should increment j by 2 instead of by 1.

9.

This question involves reasoning about a simulation of balls being thrown at a target. The simulation is encapsulated in the following TargetSimulation class. You will write two of the methods in this class.

Class information for this question

```
public class TargetSimulation
{
        /** Radius from the center of the target to edge of the  target. */
        private int targetRadius;

        /** Maximum number of balls to be thrown at the target in each
            simulation. */
        private int maxBalls;

        /** Constructs a TargetSimulation where tar is the target radius and
        * max is the maximum number of balls to throw at the target.
        * Precondition: tar > 0; max > 0
        */
        public TargetSimulation(int tar, int max)
        {
                targetRadius = tar;
                maxBalls = max;
        }

        /** Returns an integer representing the distance from the center of
        * the target to the location where a ball hits or passes the target.
        */
        private int throwAccuracy()
        { /* implementation not shown */ }

        /** Simulates balls being thrown at a target.
        * Returns true if a ball hits the target; false otherwise.
        */
        public boolean simulate()
        { /* to be implemented in part (a) */ }

        /** Runs num simulations and returns the proportion of simulations
        * in which a ball hits the target.
        * Precondition: num > 0
        */
        public double runSimulations(int num)
        { /* to be implemented in part (b) */ }
}
```

(a) Write the simulate method, which simulates balls being thrown at a target. The method returns true if a ball hits the target within a maximum number of throws, otherwise the method returns false.

The TargetSimulation class provides a method called throwAccuracy() that returns an integer representing the distance from the center of the target to where a ball hits or misses the target. The returned distance may vary from call to call.
Balls are thrown at the target until one of the following conditions becomes true:

- A ball has hit the target.
- The maximum number of balls has been thrown.

Complete method simulate below. You must use hopDistance appropriately to receive full credit.

```
/** Simulates balls being thrown at a target.
  * Returns true if a ball hits the target; false otherwise.
*/
public boolean simulate()
```

(b) Write the runSimulations method, which performs a given number of simulations and returns the proportion of simulations in which a ball has hit the target. For example, if the parameter passed to runSimulations is 400, and 100 of the 400 simulate method calls returned true, then the runSimulations method should return 0.25.

Complete method runSimulations below. Assume that simulate works as specified, regardless of what you wrote in part (a). You must use simulate appropriately to receive full credit.

```
/** Runs num simulations and returns the proportion of simulations in which a
* ball hits the target.
* Precondition: num > 0
*/
public double runSimulations(int num)
```

10.

This question involves the implementation of a sleep tracking system that is represented by the SleepTracker class. A SleepTracker object is created with parameters that defines the minimum and maximum hours of sleep.
The SleepTracker class provides a constructor and the following methods.

- `addHoursSlept`, which records the hours slept in one night.
- `totalNights`, which returns the total number of nights.
- `under`, which returns the fraction of nights with hours below the minimum.
- `over`, which returns the fraction of nights with hours above the maximum.

The following is a sample of code which uses the `SleepTracker` class:

```
SleepTracker st = new SleepTracker( 8, 10 );
// sleepfoundation.org recommends 8-10 hours of sleep for teens

System.out.println( st.totalNights()+“, “+st.under()+“, “+st.over() );

st.addHoursSlept( 8.5 );
System.out.println( st.totalNights()+“, “+st.under()+“, “+st.over() );

st.addHoursSlept( 4.0 );
st.addHoursSlept( 4.5 );
System.out.println( st.totalNights()+“, “+st.under()+“, “+st.over() );

st.addHoursSlept( 10.1 );
System.out.println( st.totalNights()+“, “+st.under()+“, “+st.over() );
```

The following lines are printed by this sample code:

```
0, 0.0, 0.0

1, 0.0, 0.0

3, 0.6666666666666666, 0.0

4, 0.5, 0.25
```

Write the complete SleepTracker class, including the constructor and any required instance variables and methods. Your implementation must meet all specifications and conform to the example.