**Question 9: TargetSimulation**

|  |  |  |
| --- | --- | --- |
| **Part (a)** | simulate | **5 points** |

**Part A**

**+1** Calls throwAccuracy() and uses the returned distance to determine whether a ball hit the target.

**+1** Initializes and accumulates the number of balls thrown and compares the count to maxBalls.

**+1** Determine if a ball has hit the target.

**+1** Return false if maxBalls has been exceeded.

**+1** Return true if a ball has hit the target.

|  |  |  |
| --- | --- | --- |
| **Part (b)** | runSimulations | **4 points** |

**+1** Calls simulate the specified number of times (no bounds errors).

**+1** Initializes and accumulates a count of true results.

**+1** Calculates proportion of successful simulations using double arithmetic.

**+1** Returns the calculated value.

Question 9: TargetSimulation

**Question 10: *SleepTracker***

**+1** Declares all appropriate private instance variables

**+2** constructor

**+1** Declares header: public SleepTracker (int \_\_\_ ).

**+1** Uses parameter and appropriate values to initialize instance variables.

**+3** addHoursSlept method

**+1** Declares header: public void addHoursSlept( int \_\_\_\_ )

**+1** Identifies nights with hours above and nights with hours below, and increments each count.

**+1** Increments the total nights count.

**+1** totalNights method

**+1** Declares and implements public int totalNights().

**+1** under method

**+1** Declares and implements public double under ().

**+1** over method

**+1** Declares and implements public double over().