22C:16 (CS:1210) Quiz 4

You have 20 minutes to complete this quiz.

1. Consider the following expressions. For each expression, first determine whether the expression is *well-formed* or not. For each well-formed expression, determine if the expression will be successfully evaluated by Python or not. For each expression that is successfully evaluated, determine its value and the type of its value. Assume that the `math`, `random`, and `sys` modules have all been imported prior to the execution of these expressions. Also, suppose that the value of `sys.maxint` is 9223372036854775807.

   (a) `len(str(random.randint(10, 20)*22))`

   (b) `-sys.maxint-1`

   (c) `5 < 8 and (100/(len("0")-1))`

   (d) `abs(5 - 25) < 150 and 0.5`

   (e) `2/math.trunc(1.5)+200L`

   (f) `5 < 8 or (100/(len("0")-1))`

   (g) `bin(len(str(10<20)))`

   (h) `4L*len("Problem1")/(len("Exam1")%3)+2.0`

   (i) `-4*(-2)**3+-len(str(66))`

   (j) `bool(math.sqrt((-1)**100))`

2. Suppose that we have defined a function called `numFactors` whose header is:

   ```
   def numFactors(n):
   ```

   The parameter `n` is expected to be a positive integer and the function returns the number of distinct factors of `n`. For example, if `n` were 10, the function would return 4 (since 10 has factors 1, 2, 5, and 10).

   For this problem, we want you to write a function called `numPrimes` with the following function header:

   ```
   def numPrimes(L, U):
   ```

   You can assume that the parameters `L` and `U` are positive integers and `L` is no greater than `U`. This function should return the number of primes that are between `L` and `U`, inclusive of `L` and `U`. This function should call `numFactors` repeatedly to complete its task.

   Here is an example that will help you understand what is being asked of you.
   Example. The function call `numPrimes(5, 10)` should return 2 because there the two primes in the range $[5, 10]$, namely 5 and 7.