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 `hotpoLength` whose header is:

```
def hotpoLength(n):
```

The parameter `n` is expected to be a positive integer and the function returns the HOTPO length of `n`. (Recall that the *HOTPO length* of a positive integer n is the number of steps that the HOTPO process takes to reach 1, starting at n.) For example, if `n` were 6, the function would return 8 (since the HOTPO process starting at 6 visits 3, 10, 5, 16, 8, 4, 2, 1).

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

```
def averageHotpoLength(M):
```

You can assume that the parameter `M` is a positive integer. This function returns the average HOTPO length for all integers 1 through M. This function should call `hotpoLength` repeatedly to complete its task.