STAT/CS 94 Fall 2015 Adhikari HW05, Due: 9/30/15

NAME: SID:

Problem 1 *Tricky Tests*

A series of multiple choice tests consists of several hundred questions. Each question has 5 possible answers, only one of which is correct. A student's total score is calculated as follows: four points are awarded for each correct answer, and one point is deducted for each answer that is wrong or missing or anything but the correct choice.

A student guesses each answer at random, independently of all the other answers. Though I'm sure you would never take a test this way, the setting is important for those who run multiple choice exams. They have to control the chance that students pass by just throwing darts at the choices.

(a) Let n be the total number of questions in the series. If possible, find the chance that the student gets the first three answers right. If this is not possible, explain why not.

(b) What is your best guess for the student's score on the test? Why?

(c) Let R be the number of questions the student gets right, and S the student's score on the test. Find a formula for S in terms of R and n. What are the possible values of R? What are the possible values of S?

Problem 2 *Lively Loops*

The Fibonacci sequence starts with the terms 1, 1. Each subsequent term is formed by adding the previous two terms in the sequence. Thus the first five terms of the sequence are 1, 1, 2, 3, and 5. Write a definition for a function named `fibonacci` that takes a single argument, a positive integer named `n`. The function should return a table of length `n` containing a single column named `Terms`, and that column should contain the first `n` Fibonacci numbers. So, for example, `fibonacci(4)` should have value `Table([np.array([1, 1, 2, 3])], ['Terms'])`.

Problem 3 *Spiffy Sampling*

The table Ages consists of just one column, labeled Age. The column contains the ages of all the people in a city that has a population of 832,304. As usual, each row of the table corresponds to one person.

Students in a data science class create Med, an empty table with just one column. The column is labeled Medians. The students then repeat the following process 400 times: *Draw a simple random sample of size 10,000 from* Ages*, compute the median age in the sample, and append the median to the column* Medians*.*

Fill in the blanks using items from the list below. You may use the same item more than once, and you may leave items unused.

- 10,000
- bar chart
- empirical distribution
- NumPy
- city
- sample
- maximum
- 40,000
- ages
- median age
- empirical
- 832,304
- probability
- uniform
- 4,000,000
- tables
- roulette
- mean age
- 400

(a) The table Ages has ____________________ rows, and Med has ____________________ rows.

(b) The ____________________ of the ____________________ in the first sample is likely to look roughly like the distribution of ages in the city.

(c) The ____________________ distribution displayed by calling Med.hist(normed=True) is likely to look roughly like the ____________________ distribution of the ____________________ of ____________________ randomly people chosen at random from the city.