NAME: SID:

Please write your answers on the printed page, in the space provided. If you print this assignment yourself, please print it *double-sided* on a single sheet of paper. (There will be a small penalty for not following this instruction; it makes the grader's job more difficult.)

Problem 1   *Historic Histograms*

In a histogram of hours of employment of a group of students, the bins include the left endpoint but not the right. The table below shows two of the bins and the heights of the corresponding bars, which are in the *density* scale. (As a reminder, this is the kind of histogram a call to `.hist(..., normed=True)` produces.) If the two bins are combined into one, what would be the height of the bar over it?

hours	$8-10$	$10-15$
height	0.15 per hour	0.02 per hour

Problem 2   *Zany Zips*

A table called `Contact_Info` consists of one row per person. The columns contain the person's name, email address, and other contact information. One of the columns is called Address and contains the person's street address written as a single string that ends with a zip code. For example, one of the entries is "1600 Pennsylvania Avenue NW, Washington, DC 20500" and another is "367 Evans Hall #3860, University of California, Berkeley, CA 94720-3860".

(a) Write a function called `zip_code` that takes as its argument an address string such as the examples above, and returns the zip code as written in the address string. Thus `zip_code('1600 Pennsylvania Avenue NW, Washington, DC 20500')` should return '20500' and `zip_code('367 Evans Hall #3860, University of California, Berkeley, CA 94720-3860')` should return '94720-3860'.

(b) Write code that augments the table `Contact_Info` with a column called 'Zip Code' that contains the zip codes extracted from the Address column by the function `zip_code` that you created in part (a). Again, your code should cause the 'Zip Code' column to be *added* to the table named `Contact_Info`.

Hint: Since you probably didn't write your function to also work on NumPy arrays, you may want to use the `Table` method `.apply()`. Look up the syntax at data8.org/datascience/tables.html if you need to.

Problem 3   *School Shuffle*

The Registrar's Office at a school has the school's complete enrollment information. For every student, it has the complete list of classes in which the student is enrolled, and for every class, it has the complete list of enrolled students. There are 300 classes being offered at the school. The Registrar selects a probability sample of students as follows: she selects a sample of 10 of the 300 classes at random without replacement, and then for each selected class, she adds all the students in that class to the probability sample. (So if a student is in more than one of the selected classes, that student appears in the probability sample more than one time.) This method is called "cluster sampling" because each selected class is a cluster of students.

(a) Do all students have the same chance of entering the sample? Explain your answer.

(b) Describe three significant differences between the properties of the Registrar's sample and properties of a sample drawn at random without replacement from among all the students.

Problem 4   *Casino Conundrum*

There are 38 pockets in a Nevada roulette wheel: 18 red numbers, 18 black numbers, and 2 green numbers. A bet on a "split" (two pockets–you win if either one is picked) pays 17 to 1.

(a) If you bet repeatedly on a split, then in the long run roughly what proportion of bets do you expect to win? What proportion do you expect to lose?

(b) If you repeatedly bet $1 on a split, then every bet you win gives you a net gain of $17, and every bet you lose gives you a net gain of −$1. In the long run, what do you expect as your average net gain per bet? [If you have trouble figuring this out, first work with 5 bets and pretend that the sequence of results was Lose, Lose, Lose, Win, Lose. Figure out your average net gain per bet, in terms of the proportions of wins and losses.]

(c) Repeat part (b) if instead you are betting $1 each time on "red"; the bet pays 1 to 1.

(d) Repeat part (b) if instead you are betting $1 each time on a "single number" (one pocket); the bet pays 35 to 1.