Midterm Exam - DSC 10, Spring 2023

Instructions:

- This exam consists of 8 questions, worth a total of 81 points.
- Write your PID in the top right corner of each page in the space provided.
- Please write **clearly** in the provided answer boxes; we will not grade work that appears elsewhere. Completely fill in bubbles and square boxes; if we cannot tell which option(s) you selected, you may lose points.

  ◯ A bubble means that you should only **select one choice**.

  ☐ A square box means you should **select all that apply**.
- You may refer to the Reference Sheet that was provided to you. Other than that, you may not refer to any other resources or technology during the exam (including calculators).

Full Name:

PID:

Lecture Section: ◯ A (12-12:50PM) ◯ B (1-1:50PM)

Seat you are in:

By signing below, you are agreeing that you will behave honestly and fairly during and after this exam.

Signature:

Version C

Please do not open your exam until instructed to do so.

Make sure you have read the data description attached to the Reference Sheet!

Question 1 (16 pts)

a) (2 pts) What is the type of the following expression?

```
(survey
 .sort_values(by="Class Standing")
)
```

◯ int or float ◯ Boolean ◯ string
◯ list or array ◯ Series ◯ DataFrame

b) (2 pts) What is the type of the following expression?

```
(survey
 .sort_values(by="Class Standing")
 .groupby("College").count()
)
```

◯ int or float ◯ Boolean ◯ string
◯ list or array ◯ Series ◯ DataFrame

c) (2 pts) What is the type of the following expression?

```
(survey
 .sort_values(by="Class Standing")
 .groupby("College").count()
 .get("IG Followers")
)
```

◯ int or float ◯ Boolean ◯ string
◯ list or array ◯ Series ◯ DataFrame

d) (2 pts) What is the type of the following expression?

```
(survey
 .sort_values(by="Class Standing")
 .groupby("College").count()
 .get("IG Followers")
 .iloc[0]
)
```

◯ int or float ◯ Boolean ◯ string
◯ list or array ◯ Series ◯ DataFrame

e) (3 pts) What **value** does the following expression evaluate to? If you believe the expression errors, provide a one sentence explanation of why.

For example, if you think the expression evaluates to an int, don't write "int" or "the largest value in the `"IG Followers"` column", write the specific int, like "23".

Hint: You have enough information to answer the problem; don't forget to look at the data description page.

```
(survey
 .sort_values(by="Class Standing")
 .groupby("College").count()
 .get("IG Followers")
 .index[0]
)
```

f) (2 pts) Consider again the expression from part (e). Suppose we remove the piece

```
.sort_values(by="Class Standing")
```

but keep all other parts the same. Does this change the value the expression evaluates to?

◯ Yes, this changes the value the expression evaluates to.

◯ No, this does not change the value the expression evaluates to.

g) (3 pts) What **value** does the following expression evaluate to? If you believe the expression errors, provide a one sentence explanation of why.

For example, if you think the expression evaluates to an int, don't write "int" or "the largest value in the `"IG Followers"` column", write the specific int, like "23".

Hint: You have enough information to answer the problem; don't forget to look at the data description page.

```
(survey
 .sort_values(by="Class Standing")
 .groupby("College").count()
 .get("IG Followers")
 .loc[0]
)
```

Question 2 (13 pts)

Fill in the blanks below so that the expression evaluates to the most unread emails of any student that has 0 Instagram followers.

```
survey[__(a)__].__(b)__
```

a) (3 pts) What goes in blank (a)?

b) (3 pts) What goes in blank (b)? Select all valid answers.

- ☐ `get("Unread Emails").apply(max)`
- ☐ `get("Unread Emails").max()`
- ☐ `sort_values("Unread Emails", ascending=False).get("Unread Emails").iloc[0]`
- ☐ `sort_values("Unread Emails", ascending=False).get("Unread Emails").iloc[-1]`
- ☐ None of the above

Now, consider the following block of code.

```
temp = survey.groupby(["IG Followers", "College"]).max().reset_index()
revellian = temp[__(c)__].get("Unread Emails").__(d)__
```

Fill in the blanks below so that `revellian` is equal to the most unread emails of any student in Revelle, among those with 0 Instagram followers.

c) (3 pts) What goes in blank (c)?

d) (2 pts) What goes in blank (d)?

e) (2 pts) The student in Revelle with the most unread emails, among those with 0 Instagram followers, is a `"HI25"` (History) major.

Is the value `"HI25"` guaranteed to appear at least once within `temp.get("Major")`?

◯ Yes ◯ No

Question 3 (12 pts)

Currently, the `"Section"` column of `survey` contains the values `"A"` and `"B"`. However, it's more natural to think of the lecture sections in terms of their times, i.e. `"12PM"` and `"1PM"`.

Below, we've defined four functions, named `converter_1`, `converter_2`, `converter_3`, and `converter_4`. All four functions return `"12PM"` when `"A"` is the argument passed in and return `"1PM"` when `"B"` is the argument passed in. Where they potentially differ is in how they behave when called on arguments other than `"A"` and `"B"`.

Note: In the answer choices below, by `None` we mean that the function call doesn't return anything, but doesn't error either.

a) (i) (2 pts) Consider the function `converter_1`, defined below.

```
def converter_1(section):
    if section == "A":
        return "12PM"
    else:
        return "1PM"
```

When `converter_1` is called on an argument other than `"A"` or `"B"`, what does it return?

◯ `"12PM"` ◯ `"1PM"` ◯ `None` ◯ It errors

(ii) (2 pts) Consider the function `converter_2`, defined below.

```
def converter_2(section):
    times_dict = {
        "A": "12PM",
        "B": "1PM"
    }
    return times_dict[section]
```

When `converter_2` is called on an argument other than `"A"` or `"B"`, what does it return?

◯ `"12PM"` ◯ `"1PM"` ◯ `None` ◯ It errors

(iii) (2 pts) Consider the function `converter_3`, defined below.

```
def converter_3(section):
    if section == "A":
        return "12PM"
    elif section == "B":
        return "1PM"
    else:
        return 1 / 0
```

When `converter_3` is called on an argument other than `"A"` or `"B"`, what does it return?

◯ `"12PM"` ◯ `"1PM"` ◯ `None` ◯ It errors

(iv) (2 pts) Consider the function `converter_4`, defined below.

```
def converter_4(section):
    sections = ["A", "B"]
    times = ["12PM", "1PM"]
    for i in range(len(sections)):
        if section == sections[i]:
            return times[i]
```

When `converter_4` is called on an argument other than `"A"` or `"B"`, what does it return?

◯ `"12PM"` ◯ `"1PM"` ◯ `None` ◯ It errors

b) Suppose we decide to use `converter_4` to convert the values in the `"Section"` column of `survey` to times. To do so, we first use the `.apply` method, like so:

```
section_times = survey.get("Section").apply(converter_4)
```

(i) (2 pts) What is the type of `section_times`?

◯ int or float ◯ Boolean ◯ string
◯ list or array ◯ Series ◯ DataFrame

(ii) (2 pts) We'd then like to replace the `"Section"` column in `survey` with the results in `section_times`. To do so, we run the following line of code, after running the assignment statement above:

```
survey.assign(Section="section_times")
```

What is the result of running the above line of code?

◯ The `"Section"` column in `survey` now contains only the values `"12PM"` and `"1PM"`

◯ The line evaluates to a new DataFrame whose `"Section"` column only contains the values `"12PM"` and `"1PM"`, but the `"Section"` column in `survey` still contains the values `"A"` and `"B"`

◯ An error is raised, because we can't use `.assign` with a column name that already exists

◯ An error is raised, because the value after the `=` needs to be a list, array, or Series, not a string

◯ An error is raised, because the value before the `=` needs to be a string, not a variable name

PID: _____

Question 4 (6 pts)

a) (2 pts) Consider the following block of code.

```
A = survey.shape[0]
B = survey.groupby(["Unread Emails", "IG Followers"]).count().shape[0]
```

Suppose the expression `A == B` evaluates to `True`. Given this fact, what can we conclude?

◯ There are no two students in the class with the same number of unread emails.

◯ There are no two students in the class with the same number of Instagram followers.

◯ There are no two students in the class with the same number of Instagram followers, and there are no two students in the class with the same number of unread emails.

◯ There are no two students in the class with both the same number of unread emails and the same number of Instagram followers.

b) We'd like to find the mean number of Instagram followers of all students in DSC 10. One **correct** way to do so is given below.

```
mean_1 = survey.get("IG Followers").mean()
```

Another two **possible** ways to do this are given below.

```
# Possible method 1.
mean_2 = (survey.get(["Section", "IG Followers"])
                .groupby("Section").mean().get("IG Followers").mean())

# Possible method 2.
X = (survey.get(["Section", "IG Followers"])
           .groupby("Section").sum().get("IG Followers").sum())
Y = survey.groupby("Section").count().get("IG Followers").sum()
mean_3 = X / Y
```

(i) (2 pts) Is `mean_2` equal to `mean_1`?

◯ Yes
◯ Yes, if both sections have the same number of students, otherwise maybe
◯ Yes, if both sections have the same number of students, otherwise no
◯ No

(ii) (2 pts) Is `mean_3` equal to `mean_1`?

◯ Yes
◯ Yes, if both sections have the same number of students, otherwise maybe
◯ Yes, if both sections have the same number of students, otherwise no
◯ No

Question 5 (6 pts)

Teresa and Sophia are bored while waiting in line at Bistro and decide to start flipping a UCSD-themed coin, with a picture of King Triton's face as the heads side and a picture of his mermaid-like tail as the tails side.



Teresa flips the coin 21 times and sees 13 heads and 8 tails. She stores this information in a DataFrame named `teresa` that has 21 rows and 2 columns, such that:

- The `"flips"` column contains `"Heads"` 13 times and `"Tails"` 8 times.
- The `"Wolftown"` column contains `"Teresa"` 21 times.

Then, Sophia flips the coin 11 times and sees 4 heads and 7 tails. She stores this information in a DataFrame named `sophia` that has 11 rows and 2 columns, such that:

- The `"flips"` column contains `"Heads"` 4 times and `"Tails"` 7 times.
- The `"Makai"` column contains `"Sophia"` 11 times.

a) (4 pts) How many rows are in the following DataFrame? Give your answer as an integer.

```
teresa.merge(sophia, on="flips")
```

Hint: The answer is less than 200.

b) (2 pts) Let A be your answer to the previous part. Now, suppose that:

- `teresa` contains an additional row, whose `"flips"` value is `"Total"` and whose `"Wolftown"` value is 21.
- `sophia` contains an additional row, whose `"flips"` value is `"Total"` and whose `"Makai"` value is 11.

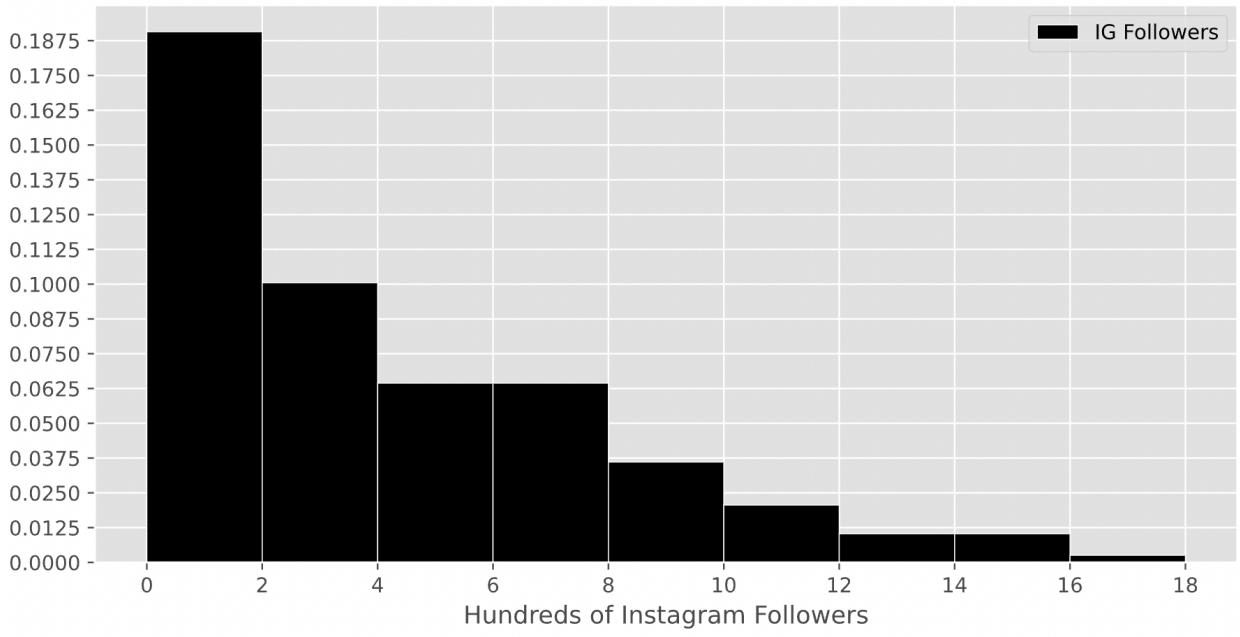
Suppose we again merge `teresa` and `sophia` on the `"flips"` column. In terms of A, how many rows are in the new merged DataFrame?

◯ A ◯ $A+1$ ◯ $A+2$ ◯ $A+4$ ◯ $A+231$

PID: _____

Question 6 (6 pts)

The histogram below displays the distribution of the number of Instagram followers each student has, measured in 100s. That is, if a student is represented in the first bin, they have between 0 and 200 Instagram followers.



For this question only, assume that there are exactly 200 students in DSC 10.

a) (4 pts) How many students in DSC 10 have between 200 and 800 Instagram followers? Give your answer as an integer.

b) (2 pts) Suppose the height of a bar in the above histogram is h. How many students are represented in the corresponding bin, in terms of h?

Hint: Just as in the first subpart, you'll need to use the assumption from the start of the problem.

◯ $20 \cdot h$

◯ $100 \cdot h$

◯ $200 \cdot h$

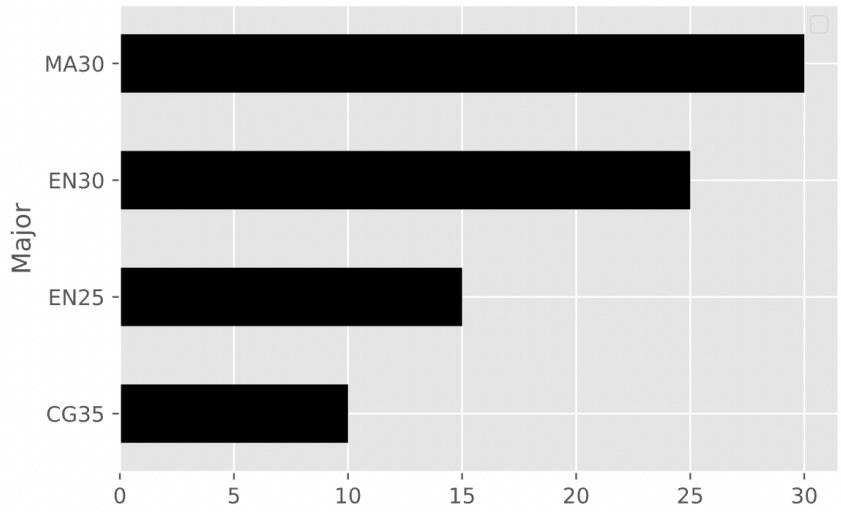
◯ $400 \cdot h$

◯ $800 \cdot h$

Question 7 (13 pts)

The four most common majors among students in DSC 10 this quarter are `"MA30"` (Mathematics - Computer Science), `"EN30"` (Business Economics), `"EN25"` (Economics), and `"CG35"` (Cognitive Science with a Specialization in Machine Learning and Neural Computation). We'll call these four majors "popular majors".

There are 80 students in DSC 10 with a popular major. The distribution of popular majors is given in the bar chart below.



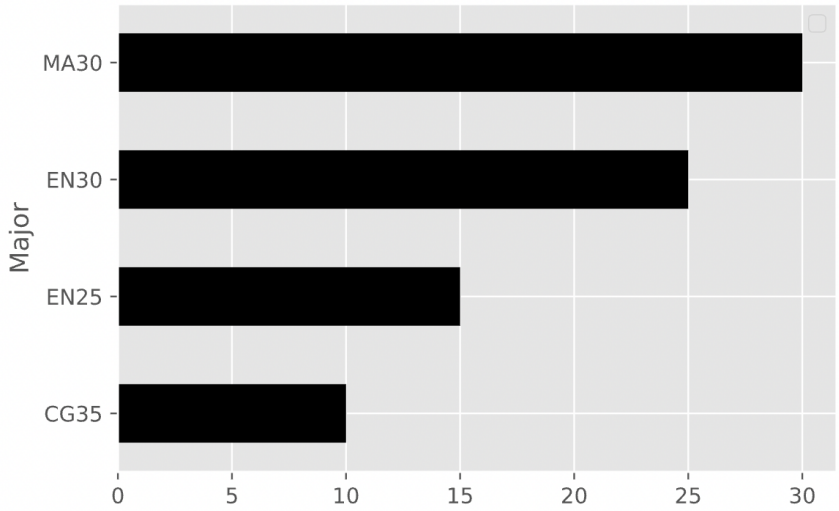
a) (3 pts) Fill in the blank below so that the expression outputs the bar chart above.

```
(survey
 .groupby('Major').count()
 .sort_values('College')
 .take(____)
 .get('Section')
 .plot(kind='barh')
)
```

What goes in the blank?

PID: _____

For your convenience, we show the bar chart from the last page again below.



b) (4 pts) Suppose we select **two** students in popular majors at random with replacement. What is the probability that both have `"EN"` in their major code? Give your answer in the form of a simplified fraction.

c) (3 pts) Suppose we select **two** students in popular majors at random with replacement. What is the probability that we select one `"CG35"` major and one `"MA30"` major (in any order)?

◯ $\frac{1}{2}$ ◯ $\frac{3}{4}$ ◯ $\frac{3}{8}$ ◯ $\frac{3}{16}$ ◯ $\frac{3}{32}$ ◯ $\frac{3}{64}$

d) (3 pts) Suppose we select k students in popular majors at random with replacement. What is the probability that we select at least one `"CG35"` major?

◯ $\frac{7}{8}$ ◯ $\frac{7^k}{8^k}$ ◯ $\frac{1}{7^k}$ ◯ $\frac{1}{8^k}$ ◯ $\frac{8^k-7^k}{7^k}$ ◯ $\frac{8^k-7^k}{8^k}$

Question 8 (9 pts)

We'd like to select three students at random from the entire class to win extra credit (not really). When doing so, we want to guarantee that the same student cannot be selected twice, since it wouldn't really be fair to give a student double extra credit.

Fill in the blanks below so that `prob_all_unique` is an estimate of the probability that all three students selected are in different majors.

Hint: The function ***`np.unique`****, when called on an array, returns an array with just one copy of each unique element in the input. For example, if* ***`vals`*** *contains the values* ***`1, 2, 2, 3, 3, 4`****,* ***`np.unique(vals)`*** *contains the values* ***`1, 2, 3, 4`****.*

```
unique_majors = np.array([])
for i in np.arange(10000):
    group = np.random.choice(survey.get("Major"), 3, __(a)__)
    __(b)__ = np.append(unique_majors, len(__(c)__))

prob_all_unique = __(d)__
```

a) (2 pts) What goes in blank (a)?

◯ `replace=True` ◯ `replace=False`

b) (2 pts) What goes in blank (b)?

c) (2 pts) What goes in blank (c)?

d) (3 pts) What could go in blank (d)? Select all that apply. At least one option is correct; blank answers will receive no credit.

☐ `(unique_majors > 2).mean()`

☐ `(unique_majors.sum() > 2).mean()`

☐ `np.count_nonzero(unique_majors > 2).sum() / len(unique_majors > 2)`

☐ `1 - np.count_nonzero(unique_majors != 3).mean()`

☐ `unique_majors.mean() - 3 == 0`

Before turning in your exam, please make sure that your PID is in the top right corner of each page.

The Data

Recall, at the start of the quarter, you were asked to complete a Welcome Survey to tell us about your background coming into the course. You were also asked to share seemingly irrelevant details, like the number of followers you have on Instagram and the number of unread emails in your primary email account. Well, those details are no longer irrelevant — in this exam, we will work with the data from this quarter's Welcome Survey!

Each row in the DataFrame `survey` represents one student in DSC 10. The information stored in the columns of `survey` are as follows:

- `"Section" (str)`: The lecture section the student is enrolled in (either `"A"` or `"B"`).
- `"IG Followers" (int)`: The number of followers the student has on Instagram. If the student does not have an Instagram account, this number is 0.
- `"Unread Emails" (int)`: The number of unread emails in the student's primary email account.
- `"College" (str)`: The college the student is a member of (either `"ERC"`, `"Marshall"`, `"Muir"`, `"Revelle"`, `"Seventh"`, `"Sixth"`, or `"Warren"`).
- `"Major" (str)`: The student's primary major, in major code form. For instance, the major code for the Data Science major is `"DS25"`.
- `"Class Standing" (str)`: The student's class standing (either `"Freshman"`, `"Sophomore"`, `"Junior"`, or `"Senior"`).

The first few rows of `survey` are shown below, though `survey` has many more rows than are pictured below (since there are more than 5 students in DSC 10 this quarter).

	Section	IG Followers	Unread Emails	College	Major	Class Standing
0	A	659	5	Revelle	CM26	Senior
1	B	814	11320	Warren	MA33	Senior
2	B	616	7340	Revelle	MA30	Junior
3	A	278	0	Seventh	HI25	Sophomore
4	A	182	4664	ERC	EN25	Senior

Throughout this exam, we will refer to `survey` repeatedly.

Assume that we have already run `import babypandas as bpd` and `import numpy as np`.