DATA 8 Summer 2023 Sample Exam.

FINAL EXAM

INSTRUCTIONS

This is your exam. Complete it either at exam.cs61a.org or, if that doesn't work, by emailing course staff with your solutions before the exam deadline.

This exam is intended for the student with email address `<EMAILADDRESS>`. If this is not your email address, notify course staff immediately, as each exam is different. Do not distribute this exam PDF even after the exam ends, as some students may be taking the exam in a different time zone.

For questions with **circular bubbles**, you should select exactly *one* choice.

- ◯ You must choose either this option
- ◯ Or this one, but not both!

For questions with **square checkboxes**, you may select *multiple* choices.

- ☐ You could select this choice.
- ☐ You could select this one too!

You may start your exam now. Your exam is due at <DEADLINE> Pacific Time. Go to the next page to begin.

Preliminaries

(a) What is your full name?

(b) What is your student ID number?

(c) Who is your lab GSI? You may write *Unknown* if you don't know their name.

(d) Sign here to confirm that all work on this exam is your own.

1. (13 points) D8 on Snackpass

The table `orders` contains information about food orders that members of Data 8 Course Staff have made this semester.

The first few rows are shown below:

User	Restaurant	Total	Receiver	Rating	With Friends
w3ndyk1m	Sharetea	5.40	stephaniekeem	10	True
s_kw33	Riceful	10.24	haileyyebonjung	2	False
nikkyp	La Burrita	14.98	wfurtaco	7	True
sonyaki55	Poke Parlor	12.86	oskibear	8	False

. . . (46 rows omitted)

The table has 6 columns:

- `User`: (string) username of the user who purchased the order
- `Restaurant`: (string) restaurant name of the order
- `Total`: (float) total amount spent on the order, in dollars
- `Receiver`: (string) username of the user who received the order
- `Rating`: (int) how the user rated their order on a scale of 1-10 (10 being most satisfied)
- `With Friends`: (boolean) whether or not the user placed the order with friends

(a) (3 points)

Complete the following line of code to visualize the relationship between how much the **order costs** versus how the **user rated** the order using a scatter plot.

```
orders._____(A)_____(_____(B)_____, _____(C)_____)
```

i. (1 pt) Fill in blank (A)

ii. (1 pt) Fill in blank (B)

iii. (1 pt) Fill in blank (C)

(b) (6 points)

Assign the variable `usually_friends` to `True` if ordering with friends is more common than not ordering with friends and `False` otherwise.

```
with_friends = orders._____(A)_____(_____(B)_____, True).num_rows
without_friends = _____(C)_____._____(D)_____ - _____(E)_____

usually_friends = with_friends _____(F)_____ without_friends
```

i. (1 pt) Fill in blank (A)

ii. (1 pt) Fill in blank (B)

iii. (1 pt) Fill in blank (C)

iv. (1 pt) Fill in blank (D)

v. (1 pt) Fill in blank (E)

vi. (1 pt) Fill in blank (F)

(c) (4 points)

Assign the variable `frugal_user` to the name of the **user** who has spent the **least** over the entire semester:

```
frugal_user = (
    orders.group(______(A)______, ______(B)______)
          .sort(______(C)______, descending=______(D)______)
          .column("User").item(0)
)
```

i. (1 pt) Fill in blank (A)

ii. (1 pt) Fill in blank (B)

iii. (1 pt) Fill in blank (C)

iv. (1 pt) Fill in blank (D)

2. (9 points) Berkeley Restaurants

For this problem we are considering restaurants around Berkeley. The `restaurant` table contains information about specific restaurants including their distance from campus in miles. **There are no duplicate restaurants in this table.**

Restaurant	Type	Distance from Campus
Round Table	Pizza	2.2
Panera	Bagels	2.3
Feng Cha	Boba	0.13
Boba Guys	Boba	1.5
Berkeley Thai House	Thai	0.15

. . . (306 rows omitted)

The `transport` table contains informations about how long it takes to get to each restaurant using various modes of transportation. **Each restaurant may appear multiple times** in this table with different modes of transportation and time in minutes.

Restaurant	Transportation	Time
Panera	Bus	27
La Burrita	Walk	5
Panera	Walk	62
Boba Guys	Drive	10
Panera	Drive	12

. . . (1492 rows omitted)

For all of the following questions you may assume you are given the function:

```
def first(some_array):
    return some_array.item(0)
```

(a) (3 pt) Which code snippet would produce a table containing the fastest transportation method to get to each restaurant?

◯
```
(transport
      .select("Restaurant", "Transportation")
      .group("Restaurant", min))
```

◯
```
(transport
      .sort("Time")
      .select("Restaurant", "Transportation")
      .group("Restaurant", first))
```

◯
```
(transport
      .sort("Time", descending=True)
      .pivot("Restaurant", first))
```

◯
```
(transport
      .sort("Time", descending=True)
      .group("Restaurant", first)
      .select("Restaurant", "Transportation"))
```

(b) (3 pt) Which code snippet would produce a table containing the fastest time for any type of food (e.g., Pizza, Bagels, Boba...)?

◯
```
(restaurant
    .join("Restaurant", transport, "Restaurant")
    .select("Type", "Time")
    .group("Type", min))
```

◯
```
(transport
    .select("Type", "Time")
    .group("Type", min))
```

◯
```
(restaurant
    .sort("Distance from Campus")
    .group("Type", first)
    .join("Restaurant first", transport, "Restaurant")
    .select("Type", "Time first"))
```

◯
```
(restaurant
    .join("Restaurant", transport, "Restaurant")
    .pivot("Type", "Time"))
```

(c) (3 pt) Which code snippet would produce a table with **columns** corresponding to each unique transportation mode (e.g., "Bus", "Drive", ...), **rows** corresponding to each unique restaurant type (e.g., Pizza, Bagels, Boba...) and the **cells** containing the **minimum travel time**.

◯
```
(restaurant
    .join("Restaurant", transport, "Restaurant")
    .pivot("Time", "Type", "Transportation", first))
```

◯
```
(restaurant
    .join("Restaurant", transport, "Restaurant")
    .select("Transportation", "Type", "Time")
    .group("Transportation", "Type", min))
```

◯
```
(restaurant
    .join("Restaurant", transport, "Restaurant")
    .pivot("Transportation", "Type", "Time", min))
```

◯
```
(restaurant
    .pivot("Transportation", "Type", "Time", min)
    .join("Restaurant", transport, "Restaurant"))
```

3. (10 points) Ski Trip

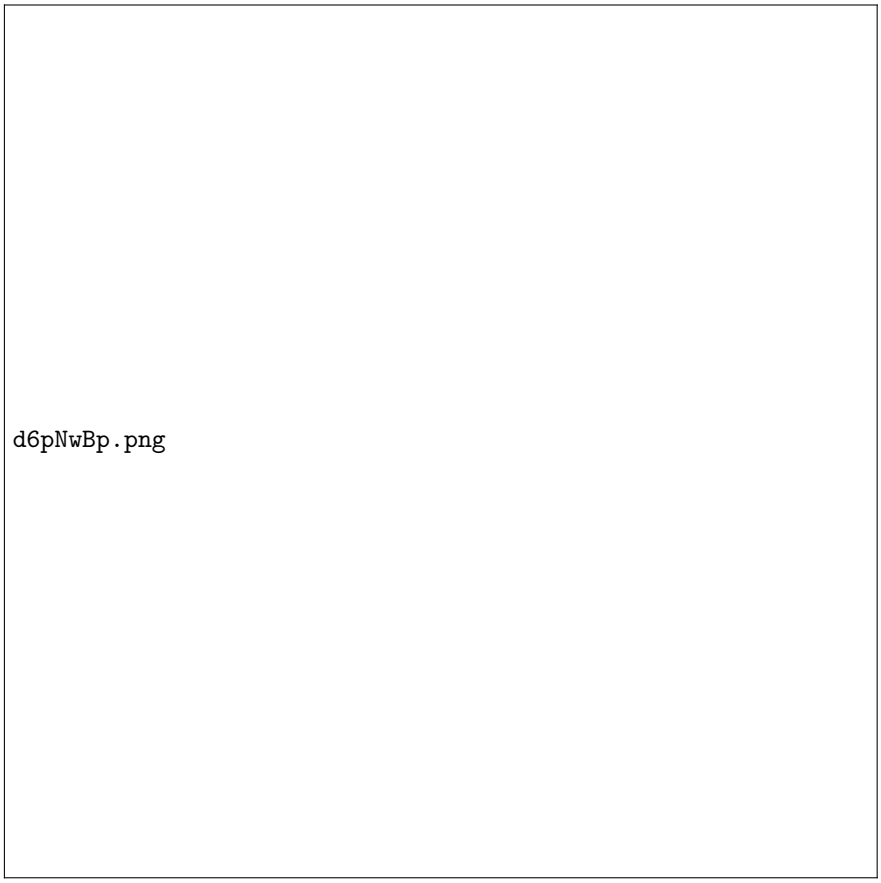
The `skiers` table below contains information about the preferences of several skiers from a **convenience sample of Data 8 staff.**

Name	Sport	Height (in)	Downhill Time (s)	Favorite Resort
James	Ski	71	90.52	Vail
Eunice	Ski	66	93.64	Beaver Creek
Oscar	Snowboard	69	89.77	Heavenly
Rebecca	Snowboard	68	91.01	Palisades
Ciara	Ski	70	101.34	Park City

. . . (40 rows omitted)

(a) (5 points)

Fill in the blanks to generate the following bar chart showing the popularity of everyone's **Favorite Resort** given in the `skiers` table. You may find the axis labels helpful.




```
favorite_counts = skiers.____(A)_____(_______(B)______)

______(C)______._____(D)______(___________(E)____________)
```

i. **(1 pt)** Fill in blank (A)

ii. **(1 pt)** Fill in blank (B)

iii. **(1 pt)** Fill in blank (C)

iv. **(1 pt)** Fill in blank (D)

v. **(1 pt)** Fill in blank (E)

(b) (3 points)

Fill in the blanks to generate the following histogram:

0kuXvGi.png

```
skiers._____(A)____(_______(B)________, group = ______(C)_______)
```

i. (1 pt) Fill in blank (A)

ii. (1 pt) Fill in blank (B)

iii. (1 pt) Fill in blank (C)

iv. (2 pt) Based on the histogram above, would it be appropriate to conclude that, **among students at UC Berkeley,** skiers are generally faster than snowboarders?

- ◯ Yes, because there is sufficient spread in the data.
- ◯ Yes, because everyone is equally likely to be a skier or snowboarder.
- ◯ No, because this is a convenience sample.
- ◯ No, because the difference in the two histograms is not big enough.

4. (15 points) Python Practice

(a) (9 points)

For each of the Python expressions below, write the output when the expression is evaluated. If the expression evaluates to an array, you should format your answer like so: `array([..., ..., ...])`. You may assume the standard imports:

```
from datascience import *
import numpy as np
```

i. (2 pt)

```
sum(make_array(1, 2, 12) >= 2)
```

ii. (2 pt)

```
make_array(2, 3, 4) - make_array(1, 2, 3)
```

iii. (3 pt)

```
x = 1
for i in make_array(3, 2, -1):
    x = x * i
print(x)
```

iv. (2 pt)

```
'data' + str(round(8.2))
```

(b) (3 pt) Which of the following functions correctly returns the number of occurrences of a specific value in a given array? For example, `count_arr_occurences(make_array(0,1,0,5,1), 1)` should evaluate to 2 and `count_arr_occurences(make_array("a", "b", "c"), "c")` should evaluate to 1.

◯
```
def count_arr_occurences(arr, value):
    count = 0
    for x in arr:
        if x == value:
            count = count + 1
    return count
```

◯
```
def count_arr_occurences(arr, value):
    return arr == value
```

◯
```
def count_arr_occurences(arr, value):
    return np.sum(arr = value)
```

◯
```
def count_arr_occurences(arr, value):
    count = 0
    for i in np.arange(value):
        if arr.item(i) == value:
            count = count + 1
    return count
```

(c) (3 pt) Which of the following will be output by running the following block of code?

```
x = 0

if x == 0:
    x = 1

if x == 1:
    x = 2
elif x < 3:
    x = 3
else:
    x = 0

print("x is", x)
```

◯ x is 0

◯ x is 1

◯ x is 2

◯ x is 3

5. (13 points) Pop vs. Rock

You want to investigate whether rock songs are less *danceable* than pop songs. A song's *danceability* is described as "...how easy it is to dance to, based on a combination of musical elements...". You collect a **random sample** of both rock and pop songs from Spotify's streaming platform and store the data in the `songs` table.

The `songs` table is shown below:

Title	Genre	Danceability
We Will Rock You	rock	42.4
Uptown Funk	pop	88.5

...(1994 rows omitted)

The table has 3 columns:

- `Title`: (string) the name of the song
- `Genre`: (string) the genre of the song which is either pop or rock
- `Danceability`: (float) a danceability rating between 0 and 100 (higher means more danceable)

(a) (3 pt) Suppose you visualize the danceability distribution for rock and pop songs by creating the following histograms using the line of code:

```
songs.hist("Danceability", group="Genre", bins=np.arange(0, 101, 5))
```

Note: All bars are visible in the histogram.

15Da0D3.png

Which of the following statements are valid conclusions, just based on the histogram above? **Select all that apply.**

☐ The most danceable song in the `songs` table was a pop song.

☐ Slightly more than 5% of pop songs had a danceability rating between 80 and 90.

☐ Roughly the same number of pop and rock songs have a danceability rating between 60 and 65.

☐ In this sample, rock and pop songs have different empirical distributions of danceability ratings.

☐ None of these.

(b) (2 pt) Suppose you want to test whether rock songs have lower danceability ratings than pop songs, on average. Which of the following is the **most appropriate null hypothesis**?

- ◯ In the population of all rock and pop songs on Spotify, rock songs have lower danceability ratings than pop songs, on average.
- ◯ In the sample, rock songs have lower danceability ratings than pop songs, on average.
- ◯ In the population of all rock and pop songs on Spotify, danceability ratings for both pop and rock songs are drawn from a uniform distribution between 0 and 100.
- ◯ In the sample, the distribution of danceability ratings is the same for pop songs as for rock songs.
- ◯ In the population of all rock and pop songs on Spotify, the distribution of danceability ratings is the same for pop songs as for rock songs.

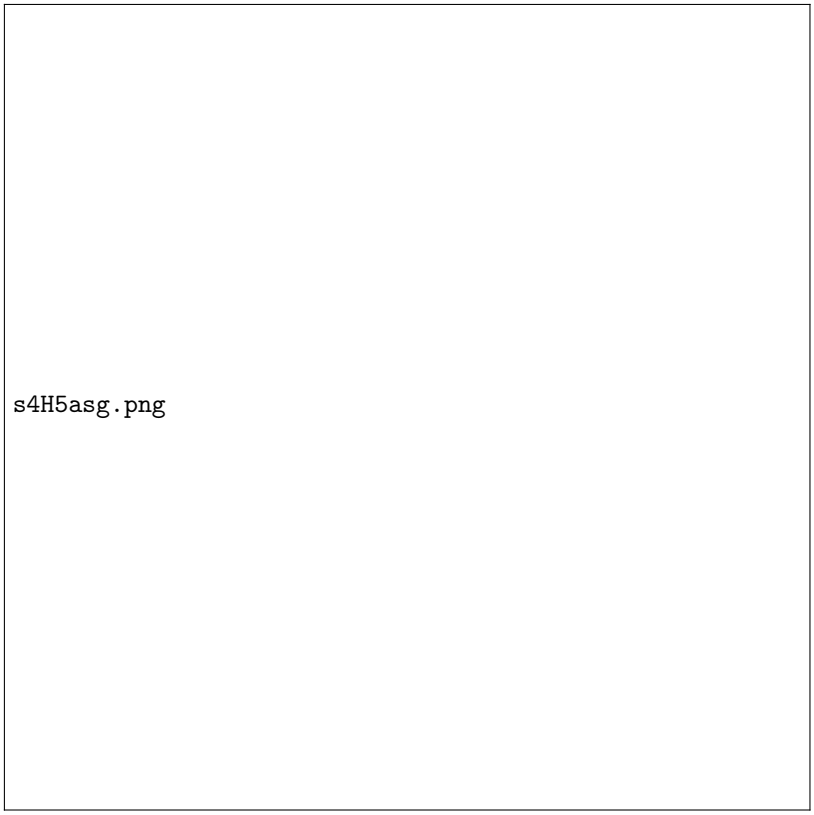
(c) (2 pt) Suppose you want to test whether rock songs have lower danceability ratings than pop songs, on average. Which of the following is the **best alternate hypothesis?**

- ◯ In the population of all rock and pop songs on Spotify, rock songs have lower danceability ratings than pop songs, on average.
- ◯ In the sample, rock songs have lower danceability ratings than pop songs, on average.
- ◯ In the population of all rock and pop songs on Spotify, danceability ratings for both pop and rock songs are drawn from a uniform distribution between 0 and 100.
- ◯ In the sample, the distribution of danceability ratings is the same for pop songs as for rock songs.
- ◯ In the population of all rock and pop songs on Spotify, the distribution of danceability ratings is the same for pop songs as for rock songs on average.

(d) (3 pt) Suppose you decide to use the difference of means between each group as your test statistic, defined as:

`average danceability rating for rock songs - average danceability rating for pop songs`

You first calculate your test statistic on your sample and save this as the `obs_stat` variable. Then, you simulate under the null hypothesis 10,000 times and record your simulated test statistics in an array called `sim_stats`. You plot both simulated and observed test statistics, as shown below:



Write a single line of code that evaluates to the empirical p-value of your test.

(e) (3 pt) Suppose you calculate your empirical p-value to be 0.003. Using a 1% p-value cutoff, which of the following are valid conclusions you can make about your test? **Select all that apply.**

☐ The data are consistent with the null hypothesis.

☐ The data are consistent with the alternative hypothesis.

☐ There is a 0.3% chance that the null hypothesis is true.

☐ If the null were true, there is a 1% chance that the null hypothesis would be incorrectly rejected.

☐ None of these.

6. (21 points) Ocean Animals

When asked to chose their favorite ocean animal out of dolphins, sea turtles, whales, and octopuses, 40% of UC Berkeley students selected dolphins, 32% selected sea turtles, 19% selected whales, and 9% chose octopuses.

The Data 8 staff selected a random sample of 500 data science majors and calculated the proportion of favorite ocean animals:

Animal	Proportion
dolphin	0.37
sea turtle	0.3
whale	0.23
octopus	0.1

We are interested in whether the distribution of favorite ocean animals for data science majors differs from the distribution of all UC Berkeley students.

(a) (6 points)

i. (2 pt) Complete the null hypothesis: *The distribution of favorite ocean animals of data science majors...*

◯ is different from the distribution of all UC Berkeley students.

◯ has exactly the same proportions as the distribution of all UC Berkeley students.

◯ is like a random sample of size 500 from a uniform distribution with a 1/4 chance for each animal.

◯ is like a random sample of size 500 from the distribution of all UC Berkeley students.

ii. (2 pt) Complete the alternative hypothesis: *The distribution of favorite ocean animals of data science majors...*

◯ is different from the distribution of all UC Berkeley students.

◯ has exactly the same proportions as the distribution of all UC Berkeley students.

◯ is like a random sample of size 500 from a uniform distribution with a 1/4 chance for each animal

◯ is like a random sample of size 500 from the distribution of all UC Berkeley students.

iii. (2 pt) Which of the following would be the best test statistic to test whether the distribution is different for data science students and UC Berkeley students? Assume that `all_students = make_array(0.4, 0.32, 0.19, 0.09)` and `ds_majors = make_array(0.37, 0.3, 0.23, 0.1)`.

◯ `np.mean(all_students) - np.mean(ds_majors)`

◯ `np.sum(all_students - ds_majors)`

◯ `np.sum(np.abs(all_students - ds_majors))`

◯ `np.mean(all_students - ds_majors)`

(b) (8 points)

Fill in the blanks below so that the code correctly performs a hypothesis test by simulating 10,000 times under the null hypothesis. Assume that we have defined `test_statistic()` correctly to compute a valid test statistic.

```
all_students = make_array(0.4, 0.32, 0.19, 0.09)
ds_majors = make_array(0.37, 0.30, 0.23, 0.10)
obs_stat = test_statistic(all_students, ds_majors)

simulated_stats = _______(A)________
for i in _______(B)________:
    one_sample = ________(C)________( ____(D)_____, all_students)
    test_stat = test_statistic(all_students, one_sample)
    simulated_stats = np.append(________(E)_______, test_stat)

p_value = np.count_nonzero(______(F)_____ >= ____(G)_____) / ____(H)____
p_value
```

i. (1 pt) Fill in blank (A)

ii. (1 pt) Fill in blank (B)

iii. (1 pt) Fill in blank (C)

iv. (1 pt) Fill in blank (D)

v. (1 pt) Fill in blank (E)

vi. (1 pt) Fill in blank (F)

vii. (1 pt) Fill in blank (G)

viii. (1 pt) Fill in blank (H)

(c) **(2 pt)** Considering that 10,000 simulations under the null hypothesis were run, which of the following is true about the p-value?

- ◯ The p-value must be 0.05 or smaller.
- ◯ According to statistical conventions, if the p-value is less than 50%, it is considered small and the result is "statistically significant."
- ◯ A p-value close to 0 means the data is consistent with the null hypothesis.
- ◯ The p-value is the probability that you conclude that the data is consistent with the null hypothesis when the alternative is actually true.
- ◯ If the p-value is exactly .001, then 10 of the simulations produced test statistics more extreme than the one observed in the data, in the direction of the alternative hypothesis.

(d) **(3 pt)** Suppose the result of running the above code leads to `p_value = 0.052`. Which of the following conclusions could be justified? **Select all that apply.**

- ☐ If we use a p-value cutoff of 5%, we should reject the null hypothesis.
- ☐ There is a 5.2% chance that the null hypothesis is true.
- ☐ Using a p-value cut-off of 5%, you can reasonably conclude that the distribution of favorite ocean animals of data science majors has the same distribution as the UC Berkeley student population.
- ☐ Using a p-value cut-off of 5%, you can reasonably conclude that the distribution of favorite ocean animals of data science majors has a different distribution as the UC Berkeley student population.
- ☐ None of these

(e) **(2 pt)** Suppose we increased the number of simulations from 10,000 to 50,000. What should we expect to happen to the p-value?

- ◯ It should be about the same (i.e., it remains pretty close to 0.0527).
- ◯ It should be about 5x larger (i.e., around 0.2635).
- ◯ It should be about 5x smaller (i.e., around 0.01054).

7. (0 points) Optional :)

(a) Draw a picture of your experience taking this exam!

(b) If there was any question on the exam that you thought was ambiguous and required clarification to be answerable, please identify the question (including the title of the section) and state your assumptions.

Note: We only plan to consider this information if we agree that the question was erroneous or ambiguous and we consider your assumption reasonable.

No more questions.