CS 171
Week 3 Lab

Overview

This lab consists of a collection of problems that you may solve in any order. To earn full credit, you must submit solutions to all problems at one time.

Important:

- You must be present in the lab session to earn credit
    - If you are recorded as absent but still complete the work, you will not earn credit. If you arrived after the CA/TA took attendance, be sure to check in so you are recorded as present/late.
    - If you miss the lab meeting, contact your instructor via email right away to request a make-up session.
- You may collaborate with at most one lab partner; Feel free to switch partners next week if you wish
- Each student must submit their own work to Gradescope for credit, even if you worked with a partner.
- At most 100 points are possible. To get a combined score, submit all of your solutions at once.
- Please only use features of the language that have been covered in reading and lectures.
    - Solutions that make use of language features not yet covered will not receive credit.
    - Students whose submissions repeatedly make use of additional skills not covered in the course so far may be suspected of violating academic integrity.
    - *Even if you know the language already, please work within the subset of the language covered up to this point in the course.*
- You may not use artificial intelligence tools to solve these problems.
- Your lab work is due at the end of the lab period. Late work will be accepted with a 30-point penalty if it is submitted within 12 hours of the end of your lab.

Helpful to Know:

- You may work these problems in any order
- Lab work will be graded by the auto-grader
- The last score earned will be recorded for your lab score
- To get a combined lab score, you must submit multiple files at the same time
- Lab assistants are here for your support. Ask them questions if you are stuck!

Materials You Are Allowed To Use:

- Lecture slides for Week 1 - 3
- Think Python Chapters 1 – 6

Important Information About The Auto-Grader

If a problem in this packet calls for you to write a function that the auto-grader checks for correctness, it is important that any code that you have written other than the function be written beneath an `if __name__ == '__main__':` block and indented.

Below is one example that *will* work and one example that *will not* work.

Correct:

```
def addTwoNumbers(num1, num2):
    sum = num1 + num2
    return sum

if __name__ == '__main__':
    a = float(input("Enter number 1: "))
    b = float(input("Enter Number 2: "))
    c = addTwoNumbers(a, b)
    print(a, "+", b, "=", c)
```

Incorrect:

```
def addTwoNumbers(num1, num2):
    sum = num1 + num2
    return sum

a = float(input("Enter number 1: "))
b = float(input("Enter Number 2: "))
c = addTwoNumbers(a, b)
print(a, "+", b, "=", c)
```

Both of these will work when you try them on your own computer, but the incorrect example will fail when the auto-grader attempts to run it.

Problem A: Museum Parking

Submission File: `lab03a.py`

The Philadelphia Museum of Art, located at the end of the iconic Benjamin Franklin Parkway, is one of the largest and most significant art museums in the United States. Established in 1876, it houses an extensive collection of over 240,000 objects, including masterpieces from the Renaissance, American, East and South Asian, Impressionist, and contemporary periods. The museum's grand architecture, reminiscent of a Greek temple, adds to its allure. Visitors can explore diverse exhibits, period rooms, and outdoor sculptures, making it a cultural treasure trove. Recent renovations led by architect Frank Gehry have enhanced the museum's accessibility and visitor experience.

While there are many ways to get to the museum, including SEPTA Bus routes 38 and 43, some visitors drive to the museum and park in the museum's parking garage. The cost to park in the garage varies based on the duration of the parking session, the day of the week, whether the person parking visits the museum, and whether the person parking has a paid membership to the museum.

Below is a screenshot from the museum's website listing the parking pricing policy.

Parking Rates

- Validate your parking ticket inside the museum at a Visitor Services Desk to receive these rates with admission.
- Parking fees must be paid using a credit card.
- **$10 Parking on Friday Nights**
  Park in the garage for just $10 with validation after 5:00 p.m. every Friday. Validate your parking ticket inside the museum at a Visitor Services Desk to receive this rate.

Public First 4 Hours	$39
Visitors First 4 Hours	$20
Members First 4 Hours First hour is FREE	$15

- Each additional hour: $2
- Per 24 hour period: $55

Write the function `museumParkingPrice(…)` with the following parameter list:

- `entryDay` – a string containing the day of arrival such as "Monday" or "Friday"

- `entryHour` – an integer between 0 and 23 representing the time of arrival
- `entryMinute` – an integer between 0 and 59 representing the time of arrival
- `exitHour` – an integer between 0 and 23 representing the time of departure
- `exitMinute` – an integer between 0 and 59 representing the time of departure
- `visitType` – a string containing either “Visitor”, “Member”, or “Public”.

The function should return an integer which represents the cost required to exit the garage.

There are some nuances to the parking policy that your solution should account for:

- A person who parks on Friday at exactly 5:00 pm (or later) is eligible for the $10 offer, but only if they are a Member or Visitor.
- A person whose exit time is equal to their entry time should not be charged. This is common if a person is unable to find parking in the garage because it is full, or their vehicle does not fit.
- Since we do not know the departure day, we can assume that a person whose exit time is earlier than their entry time stayed overnight.
- The $10 Friday offer is only good on Friday. A person who parks in the deck on Friday evening and who stays into Saturday is billed the $10 for Friday plus $2 per hour beginning at midnight on Saturday.

Examples

- `museumParkingPrice("Tuesday", 10, 15, 13, 45, "Visitor")` should return 20, as this person parked for 3 hours and 30 minutes at standard visitor pricing. Parking sessions up to 4 hours cost $20 for visitors.
- `museumParkingPrice("Wednesday", 9, 45, 13, 46, "Member")` should return 17, as this person parked for 4 hours and 1 minute at the discounted member rate. The first 4 hours of the session cost $15, and each additional hour (or portion of an hour) cost an additional $2.
- `museumParkingPrice("Sunday", 13, 45, 10, 15, "Public")` should return 55, as this person parked for 20 hours and 30 minutes at standard public pricing. The first 4 hours cost $39, and the next 16 hours and 30 minutes (which rounds up to 17 hours) would add an additional $34 to the bill. Fortunately, the most a person would pay in any single 24-hour period is $55.
- `museumParkingPrice("Friday", 18, 18, 19, 7, "Member")` should return 0, as this person is a member who parked for 1 hour or less.
- `museumParkingPrice("Friday", 18, 18, 7, 9, "Visitor")` should return 26. Friday evening’s portion of the parking session is covered by the $10 Friday special, but the additional parking hours between midnight and the 7:09am departure accumulated an additional $16.

Problem B: Income Tax

Submission File: `lab03b.py`

Everyone who earns income in the United States is expected to pay a portion of that income to the government in the form of income taxes. The information and examples in this lab exercise assume that the taxpayer is filing as a single (un-married) individual.

The U.S. has a progressive tax system: as your taxable income increases, portions of it are taxed at different rates. This means you don't pay one flat rate on your entire income. Instead, you pay a certain percentage on income in each bracket.

Bracket Ranges

Each bracket corresponds to a range of income taxed at a specific rate. For single filers in Tax Year 2024, the federal tax brackets look like this:

10%: on income from $0 to $11,600
12%: on income from $11,601 to $47,150
22%: on income from $47,151 to $100,525
24%: on income from $100,526 to $191,950
32%: on income from $191,951 to $243,725
35%: on income from $243,726 to $609,350
37%: on income at or above $609,351

Source: https://taxfoundation.org/data/all/federal/2024-tax-brackets/

Only the Amount in Each Bracket Is Taxed at That Rate

The first $11,600 of your taxable income (if you have at least that much) is taxed at 10%.
The amount from $11,601 to $47,150 is taxed at 12%.
And so on, moving up through the brackets.

Write the function `calculateIncomeTax(taxableIncome)` which determines a single person's tax obligation based on the information above. The function should round the final answer to the nearest dollar using Python's `round()` function.

Example Calculations - - Let's say the taxable income is $50,236.

- The first $11,600 of income is taxed at 10%
  10% of $11,600 is $1,160
- The next $35,550 of income is taxed at 12%
  Note: $35,550 is the difference between $47,150 and $11,600
  12% of $35,550 is $4,266
- The remaining $3,086 of income is taxed at 22%
  22% of $3,086 is $678.92
- This person's tax obligation is $6,105, which is the sum of $1,160, $4,266, and $678.92, rounded to the nearest dollar.

Example Calculation -- Let's say the taxable income is $47,275.

- The first $11,600 of income is taxed at 10%
  10% of $11,600 is $1160
- The next $35,550 of income is taxed at 12%
  Note: $35,550 is the difference between $47,150 and $11,600
  12% of $35,550 is $4,266
- The remaining $125 of income is taxed at 22%
  22% of $125 is $27.50
- This person's tax obligation is $5,454.00, which is the sum of $1,160, $4,266, and $27.50, rounded up to the nearest dollar.
  Note: Python's `round()` function uses "Banker's Rounding"

Example Calculation -- Let's say the taxable income is $113,900.

- The first $11,600 of income is taxed at 10%
  10% of $11,600 is $1,160
- The next $35,550 of income is taxed at 12%
  Note: $35,550 is the difference between $47,150 and $11,600
  12% of $35,550 is $4,266
- The next $53,375 of income is taxed at 22%
  Note: $53,375 is the difference between $100,525 and $47,150
  22% of $53,375 is $11,742.50
- The remaining $13,375 of income is taxed at 24%
  24% of $13,375 is $3,210
- This person's tax obligation is $20,378.00, which is the sum of $1,160, $4,266, $11,742.50 and $3,210, rounded down to the nearest dollar.
  Note: Python's `round()` function uses "Banker's Rounding"

Problem C: Find the Median

Submission File: `lab03c.py`

Write a program that takes in three integers and outputs the median value.

The median is found by ordering all data points and picking out the one in the middle.

Ex: If the input is:

```
7
4
1
```

the output is:

```
4
```

For this problem, you will read input from the keyboard. Do not display any prompt text; assume the user knows to enter three integer values. You are not responsible for detecting or correcting invalid inputs.

While you may choose to write a function to solve this problem, that is not required. Be sure to use only the features of the language which have been covered so far in the course.