Sami Case

CSC-500 Principles of Programming

Module 3 Critical Thinking Assignment

**Pseudocode**

**Part 1**

1. Import the math library.
2. Prompt the user to enter the total amount for food charges and store it as food\_total.
3. Set tax\_rate to 0.07 (representing a 7% tax rate). Can be modified in the code, but not something that usually changes at a restaurant often.
4. Prompt the user to enter the desired tip percentage, divide it by 100, and store it as tip\_rate. This depends on what the user wants to do.
5. Calculate the tax:
   1. Multiply food\_total by tax\_rate.
   2. Round up the result to the nearest hundredth (using math.ceil functions) to avoid fractions of a cent.
   3. Store this result as tax.
6. Calculate the tip:
   1. Multiply food\_total by tip\_rate.
   2. Round up the result to the nearest hundredth (using math functions) to avoid fractions of a cent.
   3. Store this result as tip.
7. Calculate the total charge:
   1. Add food\_total, tax, and tip.
   2. Store this sum as total\_charge.
8. Display:
   1. "Food Total:" followed by food\_total, formatted to two decimal places.
   2. "Sales Tax (X%):" where X is the tax rate multiplied by 100, followed by tax, formatted to two decimal places.
   3. "Service Tip (Y%):" where Y is the tip rate multiplied by 100, followed by tip, formatted to two decimal places.
   4. "Total Charge:" followed by total\_charge, formatted to two decimal places.
9. Display a “Thank-you” message to be nice.

**Part 2**

1. Ask the user to enter the current time (in hours on a 24-hour clock) and store this as current\_time.
2. Ask the user to enter the number of hours to wait for the alarm and store this as wait\_hours.
3. Calculate the time when the alarm will go off by adding current\_time and wait\_hours, then taking the remainder when divided by 24. Store this result as alarm\_time.
4. Display the message: "The time when the alarm goes off will be:" followed by alarm\_time.

**Source Code**

**Part 1**

import math

#Ask the user for the food charge.

food\_total = float(input('Enter the total amount for food charges:'))

#Calculate tax & tip

tax\_rate = 0.07 #Tax rate is 7%.

tip\_rate = float(input('What percentage (%) would you like to tip for service?'))/100 #User can decide how much to tip. In this ex., 18%.

tax = math.ceil(food\_total \* tax\_rate \* 100.0)/100.0 #You can't have a fraction of a cent, so this allows us to always round up to the 100th place.

tip = math.ceil(food\_total \* tip\_rate \* 100.0)/100.0 #You can't have a fraction of a cent, so this allows us to always round up to the 100th place.

#Calculate

total\_charge = food\_total + tax + tip

print('Food Total: ${:.2f}'.format(food\_total))

print('Sales Tax ({:.0f}%): ${:.2f}'.format(tax\_rate\*100,tax))

print('Service Tip ({:.0f}%): ${:.2f}'.format(tip\_rate\*100,tip))

print('\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_')

print('Total Charge: ${:.2f}'.format(total\_charge))

print('\nThank you!')

**Part 2**

# Ask the user for the current time (in hours on a 24-hour clock)

current\_time = int(input("Enter the current time (in hours on a 24-hour clock): "))

# Ask the user for the number of hours to wait until the alarm goes off

wait\_hours = int(input("Enter the number of hours to wait for the alarm: "))

# Calculate the time the alarm will go off

alarm\_time = (current\_time + wait\_hours) % 24

# Output the result

print('The time when the alarm goes off will be: {}'.format(alarm\_time))

**Image of Successful Execution**

**Part 1**

User inputs (10.00, 18)

A screenshot of a computer program

Description automatically generated

**Part 2**

User Inputs (13, 30)

A screen shot of a computer program

Description automatically generated

**GIT Repository**

<https://github.com/samilcase/CSC-500-Assignments>

A screenshot of a computer

Description automatically generated