**Course**: CSC500\_1

**Module & Assignment**: 3 & Critical Thinking Assignment

**Student**: Jegan Palaniyandi

### Creating Python Programs

#### **Part 1**

Write a program that **calculates the total amount of a meal** purchased at a restaurant. The program should **ask the user to enter the charge** for the food and then **calculate the amounts with an 18 percent tip and 7 percent sales tax**. **Display** each of these **amounts** and the **total price**.

**Pseudocode**

BEGIN

FUNCTION display\_total\_cost(meal\_charge)

SET calculated\_tip TO meal\_charge TIMES 0.18

SET calculated\_tax TO meal\_charge TIMES 0.07

SET total\_cost TO meal\_charge PLUS calculated\_tip PLUS calculated\_tax

PRINT "Meal charge --> $", CALL format\_dollars(meal\_charge)

PRINT "18% tip --> $", CALL format\_dollars(calculated\_tip)

PRINT "7% tax --> $", CALL format\_dollars(calculated\_tax)

PRINT "Total Cost --> $", CALL format\_dollars(total\_cost)

END FUNCTION

FUNCTION format\_dollars(amount)

RETURN formatted amount with two decimal places and right-aligned

END FUNCTION

# Main function is the entry point of the program

FUNCTION MAIN

SET user\_charge TO USER INPUT as float

IF user\_charge LESS THAN 10 OR GREATER THAN 30 THEN

PRINT "Entered charge is out of range. Please enter the correct charge."

TERMINATE PROGRAM

END IF

CALL display\_total\_cost(user\_charge)

END

**Coding**

# This function calculates and prints the total cost including tip and tax

def display\_total\_cost(meal\_charge):

# 18 percent tip

# 7 percent sales tax

calculated\_tip = meal\_charge \* 0.18

calculated\_tax = meal\_charge \* 0.07

# Total Cost = meal charge + 18% tip + 7% Sale tax

total\_cost = meal\_charge + calculated\_tip + calculated\_tax

print(f'\nMeal charge --> $ {format\_dollars(meal\_charge)}')

print(f' 18% tip --> $ {format\_dollars(calculated\_tip)}')

print(f' 7% tax --> $ {format\_dollars(calculated\_tax)}')

print('-' \* 24)

print(f' Total Cost --> $ {format\_dollars(total\_cost)}')

# This function formats the given amount for equal indentation in the receipt

def format\_dollars(amount):

# format amounts to 6 places with first three places for whole number

# and last three places for .(dot) and two fractional numbers

formatted\_number = "{: >6.2f}".format(amount)

return formatted\_number

if \_\_name\_\_ == '\_\_main\_\_':

print()

print('\*' \* 65)

print(' ' \* 14 + 'MODULE THREE ASSIGNMENT - PART ONE')

print(' ' \* 16 + '(MEALS TOTAL COST CALCULATOR)')

print('\*' \* 65)

# Get the meal charge from user

user\_charge = float(input('\nEnter the meal charge: '))

# Validate the user charge entered

if user\_charge < 10 or user\_charge > 30:

print('Entered charge is out of range with average price. Please re-run the program and enter correct charge.')

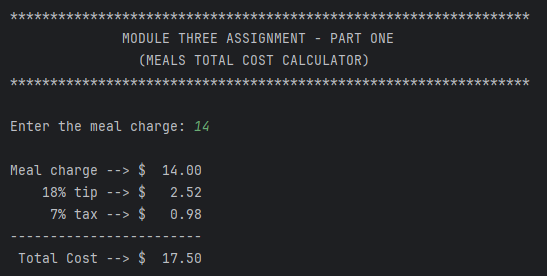
exit(4)

# Calculate and display the receipt including tax and tips

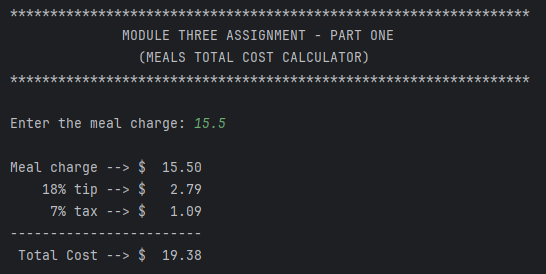
display\_total\_cost(user\_charge)

**Outputs**

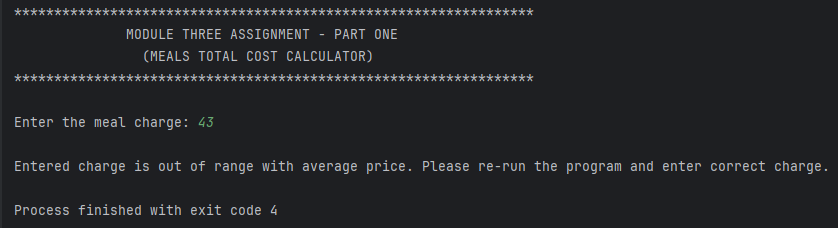
**User charge = 14**



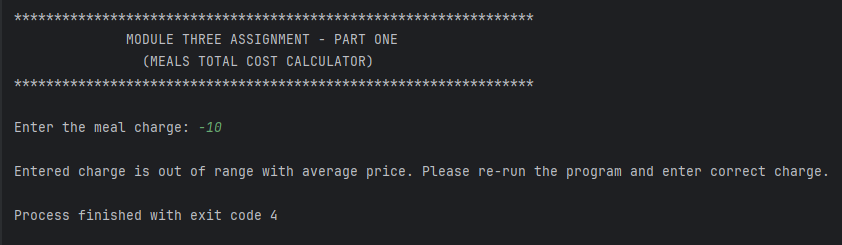
**User Charge = 15.5**



**User Charge = 43**



**User Charge = -10**



### 

### Creating Python Programs

#### **Part 2**

Many people keep time using a **24-hour clock** (11 is 11am and 23 is 11pm, 0 is midnight). If it is **currently 13** and you **set your alarm to go off in 50 hours**, it will be **15** (3pm). Write a Python program to solve the general version of the above problem. **Ask the user for the time now (in hours)** and then **ask for the number of hours to wait** for the alarm. Your program should **output what the time will be on a 24-hour clock when the alarm goes off**.

**Pseudocode**

BEGIN

FUNCTION calculate\_hour\_of\_alarm(curr\_hour, hours\_from\_now)

SET alarm\_hour TO (curr\_hour PLUS hours\_from\_now) MODULO 24

PRINT "Hour in a 24 hour clock when the alarm goes off -->", alarm\_hour

SET meridiem TO EMPTY STRING

IF alarm\_hour GREATER THAN 12 THEN

SET meridiem TO "PM"

ELSE

SET meridiem TO "AM"

END IF

PRINT "Hour in a 12 hour clock when the alarm goes off -->", alarm\_hour MODULO 12, meridiem

END calculate\_hour\_of\_alarm FUNCTION

# Main function is the entry point of the program

MAIN FUNCTION

PRINT "Welcome to the Alarm Time Hour Calculator"

SET current\_hour TO USER INPUT as integer

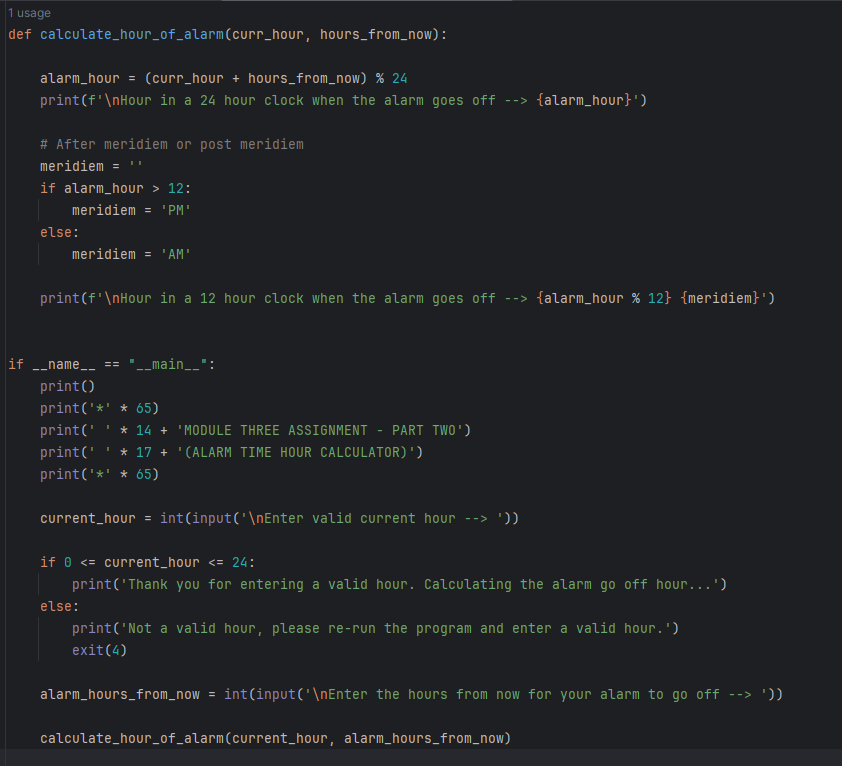
SET alarm\_hours\_from\_now TO USER INPUT as integer

CALL calculate\_hour\_of\_alarm(current\_hour, alarm\_hours\_from\_now)

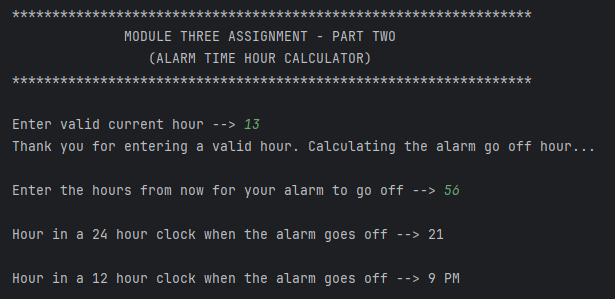
END MAIN FUNCTION

END

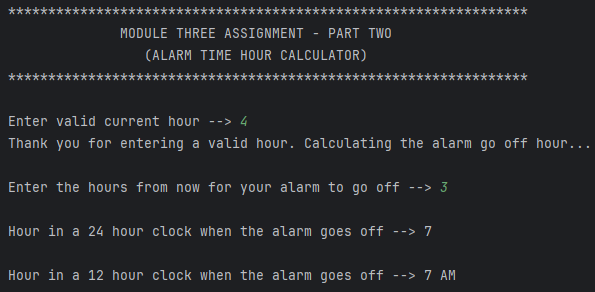
**Coding**

****

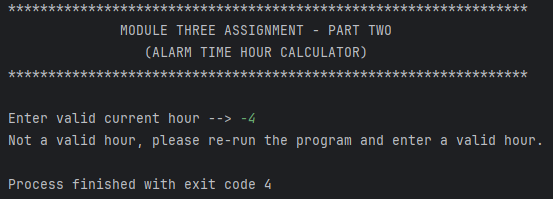
**Output One**

****

**Output Two**

****

**Output Three**

****