**Discussion 1**

You have a dataset containing the recorded daily temperatures for one month. Your task is to determine the number of days within that month where the temperature exceeded 25 degrees Celsius. Briefly describe the steps to Solve the Problem.

1. Define the variable “month” as int to value 30 or 31
2. Define the variable “numex25” as int to value 0
3. Enter values of daily temperatures in array temp = [day1, day2, day3,…,day30]. If necessary, use input with repetition for x in range(month). The input should be int(input(“Temperature day “, x+1, “: “).
4. Once all data is entered, check each individual data using for loop.

for x in range(month). Check if data day x+1 (index x) is more than 25, then add variable numex25 by 1 by numex25 += 1

1. Print the numex25 variable, i.e. print(“Number of days temp exceeding 25 C: “, numex25)

Pseudocode can be written like this:

Define month to 30

Define numex25 to 0

Define temp array to []

For x in range(month):

Request user input temp

Append user input temp to temp array

If user input temp is more than 25:

Add numex25 by 1

Print numex25

**Discussion 2**

What is the output of the following Python program?

value = 6

if value % 2 == 0:

print("first", value)

elif value % 3 == 0:

print("second", value)

while value <= 9:

value = value + 1

if value == 8:

continue

else:

pass

print ("third", value)

else:

print ("fourth", value)

print("fifth", value)

The output is

first6

third7

third9

third10

fourth10

fifth10

**Discussion 3**

The following program calculates the number of input strings with letter ‘a’, and end the program when the input string is “####”. Here is an expected sample run:

***Sample :***

enter a string (enter #### to stop): apple

enter a string (enter #### to stop): banana

enter a string (enter #### to stop): strawberry

enter a string (enter #### to stop): book

enter a string (enter #### to stop): ####

3 strings with letter 'a'

while True:

str = input("enter a string: ")

for letter in str:

if letter == 'a':

break

count +=1

print(count , "strings with letter 'a'")

There are some errors in the above program. Please indicate where the errors are and how to correct them.

1. count needs to be predefined as int. It can be count = 0 before while
2. letter needs to be predefined as string. It can be letter = “” before while
3. There should be if to determine whether the string input is #### (stop) or not. for letter in str should only be executed if it isn’t. It can be arranged like this:

if str != “####”:

for letter in str:

//for content goes here

else:

break

This will break the “while” loop if it’s “####”, thus no longer requesting input from the user, without breaking the “while” loop if it isn’t.

1. break and count +=1 statements are incorrectly placed. Placing it like that will instead count the strings without ‘a’. They need to be swapped like this:

if letter == ‘a’:

count += 1

break

1. print statement is done only if while is done, so there’s no indent so it’s not part of while.

**Discussion 4**

Write a simple Python program to implement the Pseudocode of FizzBuzz problem in discussion #1.

month = 0

numex25 = 0

temp = []

while month == 0:

    reqmonth = input("Enter number of days of the month: ")

    if reqmonth.isnumeric():

        month = int(reqmonth)

    else:

        print("Hey number of months must be a NUMBER!")

for x in range(month):

    temptoinput = 0

    temptoinput\_val = False

    while temptoinput\_val == False:

        message = "Temperature day " + str(x+1) + ": "

        temptoinputreq = input(message)

        if temptoinputreq.isnumeric():

            temptoinput = int(temptoinputreq)

            temptoinput\_val = True

        else:

            print("Hey temp must be NUMBER!")

    temp.append(temptoinput)

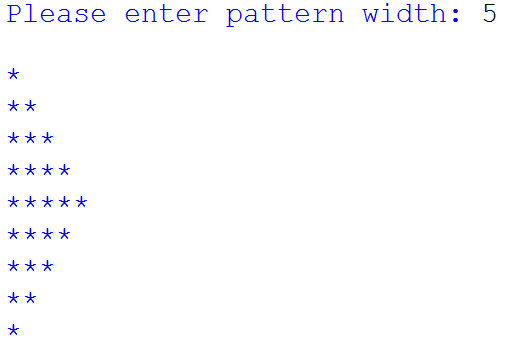
    if temptoinput > 25.0:

        numex25 += 1

print("Number of days when temp exceeds 25\*C is", numex25, "days.")

**Discussion 5**

Write a Python program that reads an integer from the user, which is the width of the pattern below, and then prints out the pattern. Suggestion: use nested **for** loops. Hint: **print("\*",end="")**.

****

Further discussion: Is it possible to use **for** only twice? Or even once? (of course no **while**)

pattern\_width = 0

while pattern\_width == 0:

    pattern\_width\_req = input("Please enter pattern width: ")

    if pattern\_width\_req.isnumeric() and int(pattern\_width\_req) != 0:

        pattern\_width = int(pattern\_width\_req)

    elif pattern\_width\_req.isnumeric():

        print("You do not want to do anything it seems.")

        break

    else:

         print("Hey pattern width must be a NUMBER!")

for y in range(pattern\_width):

    for x in range(y+1):

        print("\* ",end="")

    print('\n')

for y in range(pattern\_width-2, 0, -1):

    for x in range(y+1):

        print("\* ",end="")

    print('\n')