Dashboard / My courses / CD19411-PPD-2022 / WEEK_07-Functions / WEEK-07_CODING

Started on	Friday, 26 April 2024, 9:17 PM
State	Finished
Completed on	Friday, 26 April 2024, 9:44 PM
Time taken	27 mins 5 secs
Marks	5.00/5.00
Grade	**50.00** out of 50.00 (**100**%)
Name	DHANUSH M 2022-CSD-A

Question **1**

Correct

Mark 1.00 out of 1.00

Write a function that takes three numbers as parameters, and returns the median value of those parameters as its result.

Answer: (penalty regime: 0 %)

Reset answer

```
def median(a, b, c):
    return (a+b+c)//3
```

	Test	Expected	Got	
✓	`print(median(10, 20, 30))`	20	20	✓
✓	`print(median(60, 50, 40))`	50	50	✓
✓	`print(median(70, 90, 80))`	80	80	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **2**

Correct

Mark 1.00 out of 1.00

Write a program that reads values from the user until a blank line is entered. Display the total of all of the values entered by the user (or 0 if the first value entered is a blank line). Complete this task using recursion. Your program may not use any loops.
Hint: The body of your recursive function will need to read one value from the
user, and then determine whether or not to make a recursive call. Your function
does not need to take any arguments, but it will need to return a numeric result.

Sample Input

5
10
15
20
25

Sample Output

75

Answer: (penalty regime: 0 %)

Reset answer

```
def readAndTotal():
  value = input("")
  if not value:
    return 0
  else:
    try:
      # Convert input to float for handling decimals
      num = float(value)
      return num + readAndTotal()  # Recursive call with sum
    except ValueError:
      print("Invalid input. Please enter a number.")
      return get_total()  # Retry on invalid input

# Get the total from the user
total = readAndTotal()
print("%.0f"%total)
```

	Input	Expected	Got	
✓	5 10 15 20 25	75	75	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **3**

Correct

Mark 1.00 out of 1.00

In this exercise you will write a function that determines whether or not a password is good. We will define a good password to be a one that is at least 8 characters long and contains at least one uppercase letter, at least one lowercase letter, and at least one number. Your function should return True if the password passed to it as its only parameter is good. Otherwise it should return False. Include a main program that reads a password from the user and reports whether or not it is good. Ensure that your main program only runs when your solution has not been imported into another file.

Sample Input 1

chennai

Sample Output 1

That isn't a good password.

Sample Input 2

Chennai18

Sample Output 2

That's a good password.

Answer: (penalty regime: 0 %)

Reset answer

```python
def checkPassword(input1):
    if len(input1)<8:
        print("That isn't a good password.")
        return

    has_upper=False
    has_lower=False
    has_digit=False

    for char in input1:
        if char.isupper():
            has_upper=True
        elif char.islower():
            has_lower=True
        elif char.isdigit():
            has_digit=True

    if has_upper and has_lower and has_digit:
        print("That's a good password.")
    else:
        print("That isn't a good password.")
```

	Test	Expected	Got	
✓	`checkPassword('chennai')`	`That isn't a good password.`	`That isn't a good password.`	✓
✓	`checkPassword('Chennai18')`	`That's a good password.`	`That's a good password.`	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **4**

Correct

Mark 1.00 out of 1.00

Write a Python function sumofsquares(m) that takes an integer m returns True if m is a sum of squares and False otherwise. (If m is not positive, your function should return False.)

Here are some examples to show how your function should work.
>>> sumofsquares(41)
True

>>> sumofsquares(30)
False

>>> sumofsquares(17)
True

Answer: (penalty regime: 0 %)

Reset answer

```
from math import *

def issquare(n):
    k = int(sqrt(n))
    return(k*k == n)

def sumofsquares(m):
    if m<=0:
        return False
    i=0
    while i**2<=m:
        sqr =m-i**2
        j= int(sqr**0.5)
        if j**2 == sqr:
            return True
        i+=1
    return False

```

	Test	Expected	Got	
✓	print(sumofsquares(41))	True	True	✓
✓	print(sumofsquares(30))	False	False	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **5**

Correct

Mark 1.00 out of 1.00

Euclid was a Greek mathematician who lived approximately 2,300 years ago. His algorithm for computing the greatest common divisor of two positive integers, a and b, is both efficient and recursive. It is outlined below:

```
If b is 0 then
    eturn a
Else
    Set c equal to the remainder when a is divided by b
    Return the greatest common divisor of b and c
```

Write a program that implements Euclid's algorithm and uses it to determine the greatest common divisor of two integers entered by the user. Test your program with some very large integers. The result will be computed quickly, even for huge numbers consisting of hundreds of digits, because Euclid's algorithm is extremely efficient.

Answer: (penalty regime: 0 %)

```python
def gcd(a, b):
    if b == 0:
        return a
    else:
        return gcd(b, a % b)

def main():
    num1 = int(input())
    num2 = int(input())

    result = gcd(num1, num2)
    print(result)

if __name__ == "__main__":
    main()
```

	Input	Expected	Got	
✓	8 12	4	4	✓
✓	720 1000	40	40	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

◄ Week-07_MCQ

Jump to...

WEEK-07-Extra ►