Started on	Friday, 12 April 2024, 12:11 PM
State	Finished
Completed on	Friday, 12 April 2024, 5:37 PM
Time taken	5 hours 26 mins
Marks	5.00/5.00
Grade	**50.00** out of 50.00 (**100**%)
Name	ABINAUV R 2022-CSD-A

Question **1**

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 squa
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

```python
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:
        j_squared = m - i**2
        j = int(j_squared**0.5)
        if j**2 == j_squared:
            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 **2**

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 pas
one that is at least 8 characters long and contains at least one uppercase letter, at least one lowercase letter, and at le
number. Your function should return True if the password passed to it as its only parameter is good. Otherwise it shou
Include a main program that reads a password from the user and reports whether or not it is good. Ensure that your r
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.`	✓

Question **3**

Correct

Mark 1.00 out of 1.00

A string with parentheses is well bracketed if all parentheses are matched: every opening bracket h
closing bracket and vice versa.

Write a Python function wellbracketed(s) that takes a string s containing parentheses and returns Tr
bracketed and False otherwise.

Hint: Keep track of the nesting depth of brackets. Initially the depth is 0. The depth increases with each op
and decreases with each closing bracket. What are the constraints on the value of the nesting depth for tl
wellbracketed?

Here are some examples to show how your function should work.

```
>>> wellbracketed("22)")
False

>>> wellbracketed("(a+b)(a-b)")
True

>>> wellbracketed("(a(b+c)-d)((e+f)")
False
```

Answer: (penalty regime: 0 %)

Reset answer

```python
def wellbracketed(s):
    depth = 0
    for char in s:
        if char == '(':
            depth += 1
        elif char == ')':
            depth -= 1
            if depth < 0:
                return False
    return depth == 0


```

	Test	Expected	Got	
✓	`print(wellbracketed("22)"))`	False	False	✓
✓	`print(wellbracketed("(a+b)(a-b)"))`	True	True	✓

Question **4**

Correct

Mark 1.00 out of 1.00

A list rotation consists of taking the last element and moving it to the front. For instance, if we rotate the list
get [5,1,2,3,4]. If we rotate it again, we get [4,5,1,2,3].

Write a Python function rotatelist(l,k) that takes a list l and a positive integer k and returns the list l after l
is not positive, your function should return l unchanged. Note that your function should not change l itse
return the rotated list.

Here are some examples to show how your function should work.

```
>>> rotatelist([1,2,3,4,5],1)
[5, 1, 2, 3, 4]

>>> rotatelist([1,2,3,4,5],3)
[3, 4, 5, 1, 2]

>>> rotatelist([1,2,3,4,5],12)
[4, 5, 1, 2, 3]
```

Answer: (penalty regime: 0 %)

Reset answer

```
def rotatelist(l,k):
    n = len(l)
    k = k % n
    return l[-k:] + l[:-k]


```

	Test	Expected	Got	
✓	`print(rotatelist([1,2,3,4,5],1))`	`[5, 1, 2, 3, 4]`	`[5, 1, 2, 3, 4]`	✓
✓	`print(rotatelist([1,2,3,4,5],3))`	`[3, 4, 5, 1, 2]`	`[3, 4, 5, 1, 2]`	✓
✓	`print(rotatelist([1,2,3,4,5],12))`	`[4, 5, 1, 2, 3]`	`[4, 5, 1, 2, 3]`	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **5**

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 resu

Answer: (penalty regime: 0 %)

Reset answer

```python
def median(a, b, c):
    if a<b<c or c<b<a:
        return b
    elif b<a<c or c<a<b:
        return a
    else:
        return c
```

	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.

◄ Week-07_MCQ

Jump to...