You can view this report online at : https://www.hackerrank.com/x/tests/1727901/candidates/57163264/report

Full Name:	Instructor
Email:	muzammil.tariq@sse.habib.edu.pk
Test Name:	**CS101 - PW9 - Fall23**
Taken On:	17 Oct 2023 16:52:43 PKT
Time Taken:	6 min 23 sec/ 3810 min
Work Experience:	1 years
Invited by:	Aisha
Skills Score:	
Tags Score:	CS101 50/50 Conditional - Without Functions 30/30 PFun 10/10 Python 10/10 String 10/10 Strings 10/10 for-in loop 10/10

100%

95/95

scored in **CS101 - PW9 - Fall23** in 6 min 23 sec on 17 Oct 2023 16:52:43 PKT

Recruiter/Team Comments:

No Comments.

Question Description	Time Taken	Score	Status
Q1 Losing Your Marbles > Coding	56 sec	10/ 10	✓
Q2 Game of Sticks - Conditionals > Coding	44 sec	20/ 20	✓
Q3 The greatest - Conditionals > Coding	1 min 25 sec	10/ 10	✓
Q4 Convert Temperature to Celsius and Kelvin > Coding	13 sec	10/ 10	✓
Q5 Iterative Exponent > Coding	44 sec	15/ 15	✓
Q6 My Way > Coding	25 sec	10/ 10	✓
Q7 Reverse Characters > Coding	1 min 13 sec	10/ 10	✓
Q8 My life is potato > Coding	26 sec	10/ 10	✓

QUESTION 1

Losing Your Marbles > Coding

QUESTION DESCRIPTION

Correct Answer

Score 10

Your younger sister collected 3,000 marbles last years. She wants you to give them away to seventeen of her closest friends. If each friend receives an equal number of marbles, how many marbles would they each get? How many would you have left over?

Expected output
On the first line, your program should print the number of marbles each friend gets as a single integer.
On the second line, your program should print the number of marbles left over as a single integer.

INTERVIEWER GUIDELINES

```
marbles = 3000
friends = 17

marbles_per_friend = marbles//17    #<--- Write your formula here.
marbles_left_over = marbles%17

print(marbles_per_friend)
print(marbles_left_over)
```

CANDIDATE ANSWER

Language used: **Python 3**

```
1 marbles = 3000
2 friends = 17
3
4 marbles_per_friend = marbles//17   # <--- Write your formula here.
5 marbles_left_over = marbles % 17
6
7 print(marbles_per_friend)
8 print(marbles_left_over)
```

TESTCASE	DIFFICULTY	TYPE	STATUS	SCORE	TIME TAKEN	MEMORY USED
Testcase 0	Easy	Sample case	Success	10	0.116 sec	9.37 KB

No Comments

QUESTION 2

Correct Answer

Score 20

Game of Sticks - Conditionals › Coding

CS101 | Conditional - Without Functions

QUESTION DESCRIPTION

Problem

In demonstrating your educational software to your younger cousins, you ask each of them to get three sticks from the garden and make a triangle with them. Of course, you know that not every set of 3 sticks can form a triangle. A triangle can only be formed if the length of every stick is less than or equal to the sum of the lengths of the other 2 sticks. Alternatively, if any stick is longer than the sum of the other 2 sticks, a triangle cannot be formed. (Triangle Inequality Theorem)

Write a program that takes as input three lengths `a`, `b`, and `c`, and prints `True` if a triangle is possible, otherwise it prints `False`.

Sample

Input	Output

3 4 5	True
3 4 15	False

Constraints

- `a` `b` `c` are positive *integers*.

INTERVIEWER GUIDELINES

Solution

```
a = int(input())
b = int(input())
c = int(input())

if c > (a+b) or b > (a+c) or a > (b+c):
    print(False)
else:
    print(True)
```

CANDIDATE ANSWER

Language used: **Python 3**

```
# Enter your code here.
a = int(input())
b = int(input())
c = int(input())

if c > (a+b) or b > (a+c) or a > (b+c):
    print(False)
else:
    print(True)
```

TESTCASE	DIFFICULTY	TYPE	STATUS	SCORE	TIME TAKEN	MEMORY USED
Testcase 0	Easy	Sample case	Success	5	0.0904 sec	9.38 KB
Testcase 1	Easy	Sample case	Success	5	0.0472 sec	9.51 KB
Testcase 2	Easy	Hidden case	Success	5	0.0476 sec	9.43 KB
Testcase 3	Easy	Hidden case	Success	5	0.0511 sec	9.63 KB

No Comments

QUESTION 3

Correct Answer

Score 10

The greatest - Conditionals › Coding CS101 Conditional - Without Functions

QUESTION DESCRIPTION

Problem

Write a program that takes three numbers, `a`, `b`, and `c` as input, and prints the largest of them all.

Sample

Input	Output
1 2 3	3
2 3 1	3
3 3 2	3

Constraint

`a`, `b`, and `c` are all real numbers.

INTERVIEWER GUIDELINES

Solution

```
a = int(input())
b = int(input())
c = int(input())
print(max(a,b,c))
```

CANDIDATE ANSWER

Language used: **Python 3**

```
# Enter your code here.
a = int(input())
b = int(input())
c = int(input())
print(max(a,b,c))
```

TESTCASE	DIFFICULTY	TYPE	STATUS	SCORE	TIME TAKEN	MEMORY USED
Testcase 0	Easy	Sample case	Success	1	0.0684 sec	9.6 KB
Testcase 1	Easy	Sample case	Success	1	0.0463 sec	9.46 KB
Testcase 2	Easy	Sample case	Success	1	0.0524 sec	9.53 KB
Testcase 3	Easy	Hidden case	Success	1	0.0375 sec	9.36 KB
Testcase 4	Easy	Hidden case	Success	1	0.0325 sec	9.41 KB
Testcase 5	Easy	Hidden case	Success	1	0.097 sec	9.3 KB
Testcase 6	Easy	Hidden case	Success	1	0.0479 sec	9.41 KB
Testcase 7	Easy	Hidden case	Success	1	0.0726 sec	9.32 KB
Testcase 8	Easy	Hidden case	Success	1	0.0795 sec	9.26 KB
Testcase 9	Easy	Hidden case	Success	1	0.0335 sec	9.43 KB

No Comments

QUESTION 4

Correct Answer

Score 10

Convert Temperature to Celsius and Kelvin › Coding

QUESTION DESCRIPTION

Problem

Write a function called `convert_temp` that takes as argument, temperature in Fahrenheit and and uses two helper functions to convert this temperature to Celsius and Kelvin respectively.

Write a helper function called `convert_to_celsius()` that takes as argument temperature in Fahrenheit and prints the temperature in Celsius (rounded off to two decimal places) using the formula:

```
C = (F - 32) / 1.8
```

Write another helper function called `convert_to_kelvin()` that takes as argument temperature in Fahrenheit and prints the temperature in Kelvin (rounded off to two decimal places) using the formula:

```
K = ((F - 32) / 1.8) + 273.15
```

Use these two functions within your `convert_temp()` function to display (i.e., print) the temperatures for the user.

Sample

```
Input: 32

Output:
The temperature in Celsius is: 0.0
The temperature in Kelvin is: 273.15
```

INTERVIEWER GUIDELINES

```
def convert_to_celsius(F):
    C = (F - 32) / 1.8
    print("The temperature in Celsius is:", round(C, 2))

def convert_to_kelvin(C):
    K = ((F - 32) / 1.8) + 273.15
    print("The temperature in Kelvin is:", round(K, 2))

def convert_temp(F):
    convert_to_celsius(F)
    convert_to_kelvin(F)
```

CANDIDATE ANSWER

Language used: **Python 3**

```
1 def convert_to_celsius(F):
2     C = (F - 32) / 1.8
3     print("The temperature in Celsius is:", round(C, 2))
4
5 def convert_to_kelvin(C):
6     K = ((F - 32) / 1.8) + 273.15
7     print("The temperature in Kelvin is:", round(K, 2))
8
9 def convert_temp(F):
```

```
    convert_to_celsius(F)
    convert_to_kelvin(F)

```

TESTCASE	DIFFICULTY	TYPE	STATUS	SCORE	TIME TAKEN	MEMORY USED
Testcase 0	Easy	Sample case	Success	2.5	0.0383 sec	9.36 KB
Testcase 1	Easy	Sample case	Success	2.5	0.0365 sec	9.43 KB
Testcase 2	Easy	Hidden case	Success	2.5	0.0323 sec	9.63 KB
Testcase 3	Easy	Hidden case	Success	2.5	0.0311 sec	9.31 KB

No Comments

QUESTION 5

Correct Answer

Score 15

Iterative Exponent › Coding

QUESTION DESCRIPTION

Problem

Write a function named `power` to *iteratively* compute the exponentiation of its parameters m and n i.e. m^n using multiplication only.

Sample

```
>>> power(3,4)
81
>>> power(0,2)
0
>>> power(12.6, -4)
Error: bad argument. power is defined for integers only.
>>> power(-2,1)
-2
```

Constraints

None. Write appropriate *guardians* in your function.

Hint

Use the type function to check the type of a value/variable to confirm if it is the correct type or not. For Example:

```
>> type(3) == int              # This will return True as 3 is an integer
True

>> type('3') == int            # This will return False as '3' is of type
string
False

>> type(True) == bool          # This will return True as True is of type
boolean
True

>> type(3.14) != int           # This will return True as 3.14 is indeed
not of type int
False
```

INTERVIEWER GUIDELINES

Solution

```
def power(m,n):
    if not (isinstance(m, int) and isinstance(n,int)):
        print('Error: bad argument. power is defined for integers only.')
        return
    reciprocal = False
    if n < 0:
        reciprocal = True
        n = -n
    product = 1
    while n > 0:
        product *= m
        n -= 1
    if reciprocal:
        product = 1 / product
    return product
```

CANDIDATE ANSWER

Language used: **Python 3**

```
1  # Enter your code here.
2  def power(m,n):
3      if not (isinstance(m, int) and isinstance(n,int)):
4          print('Error: bad argument. power is defined for integers only.')
5          return
6      reciprocal = False
7      if n < 0:
8          reciprocal = True
9          n = -n
10     product = 1
11     while n > 0:
12         product *= m
13         n -= 1
14     if reciprocal:
15         product = 1 / product
16     return product
```

TESTCASE	DIFFICULTY	TYPE	STATUS	SCORE	TIME TAKEN	MEMORY USED
TestCase 0	Easy	Sample case	Success	2	0.0338 sec	9.34 KB
TestCase 1	Easy	Sample case	Success	2	0.0335 sec	9.3 KB
TestCase 2	Easy	Sample case	Success	2	0.0516 sec	9.29 KB
TestCase 3	Easy	Sample case	Success	3	0.034 sec	9.5 KB
TestCase 4	Easy	Sample case	Success	3	0.0752 sec	9.35 KB
TestCase 5	Easy	Sample case	Success	3	0.0537 sec	9.36 KB

No Comments

QUESTION 6

Correct Answer

My Way › Coding Strings CS101 for-in loop

QUESTION DESCRIPTION

Score 10

Problem

Write a function called `length` that takes a parameters `s` and returns the length of `s`. Do not use `len()`.

Sample

```
>>> length('Hello World')
11
>>> length('Yohsin')
6
>>> length('kilometer')
9
```

Input Format

The input contains `s` on the first line.

Constraints

- `isinstance(s, str)` is `True`

INTERVIEWER GUIDELINES

Solution

```
s = input()
def length(s):
    a = 0
    for i in s:
        a += 1
    return a
```

CANDIDATE ANSWER

Language used: **Python 3**

```
1 # Enter your code here.
2 s = input()
3 def length(s):
4     a = 0
5     for i in s:
6         a += 1
7     return a
```

TESTCASE	DIFFICULTY	TYPE	STATUS	SCORE	TIME TAKEN	MEMORY USED
TestCase 0	Easy	Sample case	Success	2	0.0579 sec	9.28 KB
TestCase 1	Easy	Hidden case	Success	4	0.0474 sec	9.21 KB
TestCase 2	Easy	Hidden case	Success	4	0.034 sec	9.52 KB

No Comments

QUESTION 7

Correct Answer

Reverse Characters › Coding

QUESTION DESCRIPTION

Score 10

Problem

Write a function `reversechar(mystr)` which reverses every two consecutive characters in given string, if the length of the string is odd the last character is reversed with a space.

Sample

```
>>> reversechar('my name is sarim')
'ymn ma esis rami'
>>> reversechar('my name is sarims')
'ymn ma esis rami s'
```

Input Format

The input contains `mystr` on the first line.

Constraints

- `isinstance(mystr, str)` is `True`

INTERVIEWER GUIDELINES

```
def reversechar(s):
    if len(s) % 2 != 0:
        s+=' '
    d = ''
    i=0
    while i<len(s):
        d +=s[i+1] + s[i]
        i +=2
    return d
```

CANDIDATE ANSWER

Language used: **Python 3**

```
# Enter your code here. Read input from STDIN. Print output to STDOUT
def reversechar(s):
    if len(s) % 2 != 0:
        s+=' '
    d = ''
    i=0
    while i<len(s):
        d +=s[i+1] + s[i]
        i +=2
    return d
```

TESTCASE	DIFFICULTY	TYPE	STATUS	SCORE	TIME TAKEN	MEMORY USED
Testcase 0	Easy	Sample case	Success	2	0.0755 sec	9.64 KB
Testcase 1	Easy	Sample case	Success	2	0.0293 sec	9.4 KB
Testcase 2	Easy	Hidden case	Success	2	0.0347 sec	9.3 KB
Testcase 3	Easy	Hidden case	Success	2	0.0454 sec	9.46 KB
Testcase 4	Easy	Hidden case	Success	2	0.0455 sec	9.38 KB

No Comments

QUESTION 8

My life is potato › Coding

Python | PFun | CS101 | String

Correct Answer

Score 10

QUESTION DESCRIPTION

The nursery rhyme "1 potato, 2 potato" consists of the following verse:

1 potato, 2 potato, 3 potato, 4!
5 potato, 6 potato, 7 potato, MORE!

Your job is to extend this song by a given number of verses, and continue counting.

For example, if you are asked to extend the song to three verses, the song will count up to 21 potatoes. See examples below.

Function Description

Write the function *potato*. The function must **return** the entire song as a string.

potato has the following parameter:

verses: an integer

Constraints

- *verses* is at least 1

Note

- HackerRank will handle input and output. **Do not read or print any values**!

▼ Input Format For Custom Testing

The first (and only) line contains an integer, *verses*, denoting the number of verses to return as string.

▼ Sample Case 0

Sample Input For Custom Testing

```
3
```

Sample Output

```
1 potato, 2 potato, 3 potato, 4!
5 potato, 6 potato, 7 potato, MORE!
8 potato, 9 potato, 10 potato, 11!
12 potato, 13 potato, 14 potato, MORE!
15 potato, 16 potato, 17 potato, 18!
19 potato, 20 potato, 21 potato, MORE!
```

Explanation

Three verses are required, consisting of 6 lines. Counting proceeds to 21.

▼ Sample Case 1

Sample Input For Custom Testing

```
10
```

Sample Output

```
1 potato, 2 potato, 3 potato, 4!
5 potato, 6 potato, 7 potato, MORE!
8 potato, 9 potato, 10 potato, 11!
12 potato, 13 potato, 14 potato, MORE!
15 potato, 16 potato, 17 potato, 18!
19 potato, 20 potato, 21 potato, MORE!
22 potato, 23 potato, 24 potato, 25!
26 potato, 27 potato, 28 potato, MORE!
29 potato, 30 potato, 31 potato, 32!
33 potato, 34 potato, 35 potato, MORE!
36 potato, 37 potato, 38 potato, 39!
```

```
40 potato, 41 potato, 42 potato, MORE!
43 potato, 44 potato, 45 potato, 46!
47 potato, 48 potato, 49 potato, MORE!
50 potato, 51 potato, 52 potato, 53!
54 potato, 55 potato, 56 potato, MORE!
57 potato, 58 potato, 59 potato, 60!
61 potato, 62 potato, 63 potato, MORE!
64 potato, 65 potato, 66 potato, 67!
68 potato, 69 potato, 70 potato, MORE!
```

Explanation

Ten verses are required, or 20 lines. Counting proceeds to 70.

INTERVIEWER GUIDELINES

```
def potato(verses):
    no = 0
    number = 1
    string = ''
    while(no<verses):
        no+=1
        string+=str(number)+' potato, ' + str(number+1) + " potato, "+
str(number+2) + " potato, "+ str(number+3)+'!\n'
        string+=str(number+4)+' potato, ' + str(number+5) + " potato, " +
str(number+6) + " potato, MORE!\n"
        number+=7
    return string
```

CANDIDATE ANSWER

Language used: **Python 3**

```
1  def potato(verses):
2      no = 0
3      number = 1
4      string = ''
5      while(no<verses):
6          no+=1
7          string+=str(number)+' potato, ' + str(number+1) + " potato, "+
8  str(number+2) + " potato, "+ str(number+3)+'!\n'
9          string+=str(number+4)+' potato, ' + str(number+5) + " potato, " +
10 str(number+6) + " potato, MORE!\n"
           number+=7
       return string
```

TESTCASE	DIFFICULTY	TYPE	STATUS	SCORE	TIME TAKEN	MEMORY USED
Testcase 0	Easy	Sample case	Success	1	0.0469 sec	9.31 KB
Testcase 1	Easy	Sample case	Success	1	0.0438 sec	9.34 KB
Testcase 2	Easy	Sample case	Success	1	0.0706 sec	9.44 KB
Testcase 3	Easy	Sample case	Success	1	0.0553 sec	9.36 KB
Testcase 4	Easy	Sample case	Success	1	0.04 sec	9.4 KB
Testcase 5	Easy	Hidden case	Success	1	0.0368 sec	9.63 KB
Testcase 6	Easy	Hidden case	Success	1	0.1221 sec	9.54 KB
Testcase 7	Easy	Hidden case	Success	1	0.0428 sec	9.32 KB
Testcase 8	Easy	Hidden case	Success	1	0.0425 sec	9.43 KB
Testcase 9	Easy	Hidden case	Success	1	0.0423 sec	9.38 KB

No Comments

PDF generated at: 17 Oct 2023 11:59:39 UTC