Rajalakshmi Engineering College

Name: Kirithick R
Email: 240701627@rajalakshmi.edu.in
Roll no: 2116240701627
Phone: 9952595005
Branch: REC
Department: I CSE FF
Batch: 2028
Degree: B.E - CSE

Scan to verify results

NeoColab_REC_CS23221_Python Programming

REC_Python_Week 4_CY

Attempt : 1
Total Mark : 40
Marks Obtained : 37.5

Section 1 : Coding

1. Problem Statement

You are tasked with designing a shipping cost calculator program that calculates the shipping cost for packages based on their weight and destination. The program utilizes different shipping rates for domestic, international, and remote destinations. The rates for each destination type are provided as global constants.

Constant Values:

DOMESTIC_RATE = 5.0

INTERNATIONAL_RATE = 10.0

REMOTE_RATE = 15.0

Function Signature: calculate_shipping(weight, destination)

Formula: shipping cost = weight * destination rate

Input Format

The first line of the input consists of a float representing the weight of the package.

The second line consists of a string representing the destinations(Domestic or International or Remote).

Output Format

The program outputs any one of the following:

1. If the input is valid and the destination is recognized, the output should consist of a single line stating the calculated shipping cost for the given weight and destination in the format: "Shipping cost to [destination] for a [weight] kg package: $[calculated cost]" with two decimal places.
2. If the input weight is not a positive float, print "Invalid weight. Weight must be greater than 0."
3. If the input destination is not one of the valid options, print "Invalid destination."

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: 5.5
Domestic
Output: Shipping cost to Domestic for a 5.5 kg package: $27.50

Answer

```
#

weight=float(input())
destination=input()
def calculation(w,d):
   if(d=="Domestic"):
      sc=w*5
   elif(d=="International"):
      sc=w*10
```

```
    elif(d=="Remote"):
        sc=w*15
    else:
        print("Invalid destination.\n")
        sc=None
    if w<0:
        print("Invalid weight. Weight must be greater than 0.\n")
        sc=None
    return sc
shipping_cost=calculation(weight,destination)

if shipping_cost is not None:
    print(f"Shipping cost to {destination} for a {weight} kg package:
${shipping_cost:.2f}")
```

Status : Correct **_Marks : 10/10_**

2. Problem Statement

Implement a program for a retail store that needs to find the highest even price in a list of product prices. Your goal is to efficiently determine the maximum even price from a series of product prices. Utilize the max() inbuilt function in the program.

For example, if the prices are 10 15 24 8 37 16, the even prices are 10 24 8 16. So, the maximum even price is 24.

Input Format

The input consists of a series of product prices separated by a space.

The prices should be entered as a space-separated string of numbers.

Output Format

If there are even prices in the input, the output prints "The maximum even price is: " followed by the maximum even price.

If there are no even prices in the input, the output prints "No even prices were found".

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 10 15 24 8 37 16

Output: The maximum even price is: 24

Answer

```
S=input()
L=S.split(" ")
l1=[]
for x in range(len(L)):
   if int(L[x])%2==0:
      l1.append(int(L[x]))
if len(l1)>0:
   print("The maximum even price is:",max(l1))
else:
   print("No even prices were found")
```

Status : Correct ***Marks : 10/10***

3. Problem Statement

Arjun is working on a mathematical tool to manipulate lists of numbers. He needs a program that reads a list of integers and generates two lists: one containing the squares of the input numbers, and another containing the cubes. Arjun wants to use lambda functions for both tasks.

Write a program that computes the square and cube of each number in the input list using lambda functions.

Input Format

The input consists of a single line of space-separated integers representing the list of input numbers.

Output Format

The first line contains a list of the squared values of the input numbers.

The second line contains a list of the cubed values of the input numbers.

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: 1 2 3

Output: [1, 4, 9]
[1, 8, 27]

Answer

```
N=list(map(int,input().split()))
Sq=list(map(lambda x: x**2,N))
Cu=list(map(lambda y: y**3,N))
print(Sq)
print(Cu)
```

Status : Correct ***Marks : 10/10***

4. Problem Statement

Create a program for a mathematics competition where participants need to find the smallest positive divisor of a given integer n. Your program should efficiently determine this divisor using the min() function and display the result.

Input Format

The input consists of a single positive integer n, representing the number for which the smallest positive divisor needs to be found.

Output Format

The output prints the smallest positive divisor of the input integer in the format: "The smallest positive divisor of [n] is: [smallest divisor]".

Refer to the sample output for the exact format.

Sample Test Case

Input: 24
Output: The smallest positive divisor of 24 is: 2

Answer

```
def spd(n):
    if n==0:
        return 0
    if n==1:
        return 1
    div=[]
    for x in range(2,int(n**0.5)+1):
        if n%x==0:
            div.append(x)
            if x*x!=n:
                div.append(n//x)
    return min(div)
N=int(input())
S_p_d=spd(N)
print(f"The smallest positive divisor of {N} is: {S_p_d}")
```

Status : Partially correct ***Marks : 7.5/10***