

WEEK 04-Iteration Control Structures-LOOPING

Started on	Wednesday, 13 March 2024, 7:30 PM
State	Finished
Completed on	Thursday, 14 March 2024, 8:37 PM
Time taken	1 day 1 hour
Marks	5.00/5.00
Grade	50.00 out of 50.00 (100%)
Name	SAKTHI MAHESWARI C 2022-CSD-A

WEEK 04-Iteration Control Structures-LOOPING

Question **1**

Correct

Mark 1.00 out of 1.00

🚩 Flag question

Determine the factors of a number (i.e., all positive integer values that evenly divide into a number).

For example:

Input	Result
20	1 2 4 5 10 20

Answer: (penalty regime: 0 %)

```
1 n=int(input())
2 for i in range(1,n+1):
3     if(n%i==0):
4         print(i,end=" ")
```

	Input	Expected	Got	
✓	20	1 2 4 5 10 20	1 2 4 5 10 20	✓
✓	5	1 5	1 5	✓
✓	13	1 13	1 13	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

WEEK 04-Iteration Control Structures-LOOPING

Question **2**

Correct

Mark 1.00 out of 1.00

🚩 Flag question

Write a program to check whether a given number is a perfect number or not.

Perfect number is a positive number which sum of all positive divisors excluding that number is equal to that number.

For example, 6 is perfect number since divisor of 6 are 1, 2 and 3.

Sum of its divisor is $1 + 2 + 3 = 6$

Sample Test Cases

Test Case 1

Input

6

Output

YES

Answer: (penalty regime: 0 %)

```
1 n=int(input())
2 s=0
3 for i in range(1,n):
4     if(n%i==0):
5         s=s+i
6 if(s==n):
7     print("YES")
8 else:
9     print("NO")
```

	Input	Expected	Got	
✓	6	YES	YES	✓
✓	45	NO	NO	✓
✓	496	YES	YES	✓
✓	123	NO	NO	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

WEEK 04-Iteration Control Structures-LOOPING

Question **3**

Correct

Mark 1.00 out of 1.00

Flag question

A number is stable if each digit occur the same number of times.i.e, the frequency of each digit in the number is the same. For e.g. 2277,4004,11,23,583835,1010 are examples for stable numbers.

Similarly, a number is unstable if the frequency of each digit in the number is NOT same.

Sample Input:

2277

Sample Output:

Stable Number

Sample Input 2:

121

Sample Output 2:

Unstable Number

Answer: (penalty regime: 0 %)

```
1 n=int(input())
2 l1=[]
3 con=0
4 for i in range(0,10):
5     c=0
6     temp=n
7     while(temp>0):
8         dig=temp%10
9         if dig==i:
10            c=c+1
11        temp=temp//10
12    if c>0:
13        l1.append(c)
14 s1=set(l1)
15 if(len(s1)==1):
16     print("Stable Number")
17 else:
18     print("Unstable Number")
19
20
```

	Input	Expected	Got	
✓	9988	Stable Number	Stable Number	✓
✓	2277	Stable Number	Stable Number	✓
✓	1233	Unstable Number	Unstable Number	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00

WEEK 04-Iteration Control Structures-LOOPING

Question 4

Correct

Mark 1.00 out of 1.00

Flag question

Strong Number:

Strong number is a special number whose sum of factorial of digits is equal to the original number.

For example: 145 is strong number. Since, $1! + 4! + 5! = 145$.

Write a program to find whether the given number is a Strong Number or not.

Input Format:

The Input consists of a single integer n.

Output Format:

Output consists of a single word 'Yes' or 'No'.

Sample Input 1:

145

Sample Output 1:

Yes

Answer: (penalty regime: 0 %)

```
1 n=int(input())
2 s=0
3 temp=n
4 while(n!=0):
5     num=1
6     fact=1
7     rem=n%10
8     while(num<=rem):
9         fact=fact*num
10        num+=1
11    s=s+fact
12    n=n//10
13 if(s==temp):
14     print("Yes")
15 else:
16     print("No")
17
```

	Input	Expected	Got	
✓	145	Yes	Yes	✓
✓	40585	Yes	Yes	✓
✓	4321	No	No	✓
✓	2	Yes	Yes	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

WEEK 04-Iteration Control Structures-LOOPING

Question **5**

Correct

Mark 1.00 out of 1.00

🚩 Flag question

Write a program to find the sum of the series $1 + 11 + 111 + 1111 + \dots + n$ terms (n will be given as input from the user and sum will be the output)

Sample Test Cases

Test Case 1

Input

4

Output

1234

Explanation:

as input is 4, have to take 4 terms.

$1 + 11 + 111 + 1111$

Answer: (penalty regime: 0 %)

```
1 n=int(input())
2 s1=int("1"*n)
3 #print(s1)
4 s=0
5 while(s1!=0):
6     s=s+s1
7     s1=s1//10
8 print(s)
9
```

	Input	Expected	Got	
✓	1	1	1	✓
✓	3	123	123	✓
✓	4	1234	1234	✓
✓	7	1234567	1234567	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.