

**Started on** Wednesday, 13 March 2024, 9:16 PM

**State** Finished

**Completed on** Thursday, 14 March 2024, 8:41 PM

**Time taken** 23 hours 25 mins

**Marks** 5.00/5.00

**Grade** 50.00 out of 50.00 (100%)

**Name** [JEESHAN R J 2022-CSD-A](#)

## Question 1

Correct

Mark 1.00 out of 1.00

**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     r=n%10
8     while(num<=r):
9         fact=fact*num
10        num+=1
11    s+=fact
12    n=n//10
13 if(s==temp):
14     print("Yes")
15 else:
16     print("No")
```

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

Question **2**

Correct

Mark 1.00 out of 1.00

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

**For example:**

Input	Result
2277	Stable Number

**Answer:** (penalty regime: 0 %)

```
1 from collections import Counter
2 def isStable(x):
3     s=str(x)
4     count=Counter(s)
5     return len(set(count.values()))==1
6 x=int(input())
7 if isStable(x):
8     print("Stable Number")
9 else:
10    print("Unstable Number")
11
```

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

Question **3**

Correct

Mark 1.00 out of 1.00

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

Test Case 2

45

Output

NO

For example:

Input	Result
6	YES

Answer: (penalty regime: 0 %)

```
1 n=int(input())
2 b=0
3 for i in range(1,n):
4     if(n%i==0):
5         b=b+i
6 if(b==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.

Question **4**

Correct

Mark 1.00 out of 1.00

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.

$1 + 11 + 111 + 1111$

Test Case 2

Input

6

Output

123456

For example:

Input	Result
3	123

Answer: (penalty regime: 0 %)

```
1 n=int(input())
2 k=int("1"*n)
3 j=0
4 while(k!=0):
5     j+=k
6     k=k//10
7 print(j)
```

	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.

## Question 5

Correct

Mark 1.00 out of 1.00

Write a [program](#) to find the count of ALL digits in a given number N. The number will be passed to the [program](#) as an input of type int.

Assumption: The input number will be a positive integer number  $\geq 1$  and  $\leq 25000$ .

For e.g.

If the given number is 292, the function should return 3 because there are 3 digits in this number

If the given number is 1015, the function should return 4 because there are 4 digits in this number

**For example:**

**InputResult**

292 3

10154

**For example:**

Input	Result
293	3

**Answer:** (penalty regime: 0 %)

```
1 n=int(input())
2 def count(n):
3     return len(str(n))
4 print(count(n))
```

	Input	Expected	Got	
✓	293	3	3	✓
✓	6788	4	4	✓