# Hacker Earth Tasks:

- You have been given 3 integers - l, r and k. Find how many numbers between l and r (both inclusive) are divisible by k. You do not need to print these numbers, you just have to find their count.
    - Input Format
- The first and only line of input contains 3 space separated integers l, r and k.
- Output Format
    - Print the required answer on a single line.

```python
In [1]:    1  def countDivisors(i,j,k):
           2      count =0
           3      for n in range(i,j+1):
           4          if n % k == 0:
           5              count += 1
           6
           7      print(count)
           8      return
           9
          10  s = input()
          11  s = s.split()
          12
          13  i = int(s[0])
          14
          15  j = int(s[1])
          16
          17  k= int(s[2])
          18
          19  countDivisors(i,j,k)
```

```
1 10 2
5
```

- You have been given a positive integer N. You need to find and print the Factorial of this number. The Factorial of a positive integer N refers to the product of all number in the range from 1 to N. You can read more about the factorial of a number here.
- Input Format:
    - The first and only line of the input contains a single integer N denoting the number whose factorial you need to find.
- Output Format
    - Output a single line denoting the factorial of the number N.

In [2]:
```python
 1  n= int(input())
 2  def factorial(n):
 3      fact =1
 4      for i in range(1,n+1):
 5          fact *= i
 6          i += 1
 7      print(fact)
 8      return
 9
10  factorial(n)
11
```

```
5
120
```

## You have been given a String S consisting of uppercase and lowercase English alphabets. You need to change the case of each alphabet in this String. That is, all the uppercase letters should be converted to lowercase and all the lowercase letters should be converted to uppercase. You need to then print the resultant String to output.

- Input Format
  - The first and only line of input contains the String S
- Output Format
  - Print the resultant String on a single line.

In [3]:
```python
 1  n=(input())
 2  print(n.swapcase())
```

```
This is python
tHIS IS PYTHON
```

In [4]:
```python
 1  #In functions
 2
 3
 4  n=(input())
 5  def toggle(n):
 6    print(n.swapcase())
 7
 8  toggle(n)
```

```
HEllO
heLLo
```

```
In [6]:   1  n= input()
          2  def a(n):
          3      for i in n:
          4          if i == i.upper():
          5              a= i. lower()
          6          elif i== i.lower():
          7              a= i.upper()
          8      print(a,end= " ")
          9  a(n)
```

HellO
o

## You have been given a String S. You need to find and print whether this string is a palindrome or not. If yes, print "YES" (without quotes), else print "NO" (without quotes).

- Input Format
  - The first and only line of input contains the String S. The String shall consist of lowercase English alphabets only.
- Output Format
  - Print the required answer on a single line.

```
In [5]:   1  s= input()
          2  def palindrome(s):
          3      if s == s[::-1]:
          4          print("YES")
          5      else:
          6          print("NO")
          7
          8  palindrome(s)
```

madam
YES

## ASCII Values

In [17]:
```
1  for i in range(0,365):
2      print(i,"-->", chr(i))
```

```
0 -->
1 --> □
2 --> □
3 --> □
4 --> □
5 --> □
6 --> □
7 --> □
8 -->
9 -->
10 -->

11 --> □
12 -->
13 -->
14 --> □
15 --> □
16 --> □
17 --> □
```

In [7]:
```
1  ord('a')
```
Out[7]: 97

In [8]:
```
1  ord('z')
```
Out[8]: 122

In [9]:
```
1  ord('A')
```
Out[9]: 65

In [10]:
```
1  ord('Z')
```
Out[10]: 90

In [12]:
```
1  chr(ord('a')-32)
```
Out[12]: 'A'

In [13]:
```
1  chr(ord('B')+32)
```
Out[13]: 'b'

```
In [25]:  1  def data(st):
          2      for i in range(0,len(st)):
          3          if(ord(st[i])>=65 and ord(st[i])<=90):
          4              print(chr(ord(st[i])+32),end='')
          5          elif(ord(st[i])>=97 and ord(st[i])<=122):
          6              print(chr(ord(st[i])-32),end='')
          7  data("Android")
          8
```

aNDROID

```
In [ ]:  1
```

```
In [ ]:   1  n=int(input())
          2  def twoStrings(s,t):
          3      f=1
          4      if len(s)!= len(t):
          5          return 'NO'
          6      else:
          7          for i in range(len(s)):
          8              if s.count(s[i]) != t.count(s[i]):
          9                  return 'NO'
         10          if f == 1:
         11              return 'YES'
         12
         13  for i in range(n):
         14      st= input().split()
         15      s=st[0]
         16      t=st[1]
         17  print(twoStrings(s,t))
```

```
In [ ]:  1
```

```
In [ ]:  1
```

```python
In [ ]:   1  n=int(input())
          2  def twostrings(s,t):
          3      f=1
          4      if len(s)!=len(t):
          5          return 'NO'
          6      else:
          7          for i in range (len(s)):
          8              if s.count(s[i])!=t.count(s[i]):
          9                  return 'NO'
         10          if f==1:
         11              return 'YES'
         12
         13
         14  for i in range(n):
         15      st=input().split()
         16      s=st[0]
         17      t=st[1]
         18      print(twostrings(s,t))
```

```
5
hello ohlel
YES
```

## Prime numbers

- You are given an integer N. You need to print the series of all prime numbers till N.
- Input Format
  - The first and only line of the input contains a single integer N denoting the number till where you need to find the series of prime number.
- Output Format
  - Print the desired output in single line separated by spaces.

```python
In [35]:   1  num=int(input())
           2  a=1
           3  def prime(num):
           4      for num in range(a,num+1):
           5          if num > 1:
           6              for i in range(2,int(num//2)+1):
           7                  if (num % i) == 0:
           8                      break
           9                  else:
          10                      print(num,end =" ")
          11  prime(num)
          12
```

```
9
2 3 5 7
```

## Char sum

- Consider All lowercase Alphabets of the English language. Here we consider each alphabet from a to z to have a certain weight. The weight of the alphabet a is considered to be 1, b to be 2, c to be 3 and so on until z has a weight of 26. In short, the weight of the alphabet a is 1, and the weight of all other alphabets is the weight of its previous alphabet + 1.
- Now, you have been given a String S consisting of lowercase English characters. You need to find the summation of weight of each character in this String.
- For example, Consider the String aba
- Here, the first character a has a weight of 1, the second character b has 2 and the third character a again has a weight of 1. So the summation here is equal to : 1+2+1=4
- Input Format:
- The first and only line of input contains the String S.

In [5]:
```python
1  s=input()
2  def sumChar(s):
3      sum=0
4      for i in range(0,len(s)):
5          if (s[i]>=chr(97)) and (s[i]<=chr(122)):
6              sum=sum+(ord(s[i])-96)
7      print(sum)
8  sumChar(s)
```

string
87

In [ ]:
```python
1
```