In [14]:

```python
# Taks-1
# input:1 10
# hint :9+1 9+3 9+5 9+7 9+9  9+11 ...
# ouput:10 12 14 16 18 20 22 24 26 28
s=int(input())
e=int(input())
temp=0
for i in range(s,e+1):
    temp=temp+1
    if(temp%2==1):
        st=9+temp
        temp=temp+1
        print(st,end=" ")
```

```
1
10
10 12 14 16 18 20 22 24 26 28
```

In [16]:

```python
# Taks-2
# input:5
# hint 5^1+1^1 5^2+2^2 5^3+3^3
# output:6 29 152 881 6250
n=int(input())
for i in range(1,n+1):
    st=n**i +i**i
    print(st,end=" ")
```

```
5
6 29 152 881 6250
```

In [24]:

```python
#Task-3
# input:10
# hint:10+1^2 9+2^3 8+3^4 7+4^5 6+5^6 5+6^7 4+7^8 ...
# output:11 17 89 ....
# output:3
#st=i+temp**temp+1
n=int(input())
temp=1
su=0
for i in range(n,1,-1):
    st=i+(temp**(temp+1))
    su=su+st%10
    temp=temp+1
#     print(st%10,end=" ")
print(su)
```

```
10
29
```

In [3]:

```
#Sum of Natural Numbers
#input:1 10
#ouput:55 -->1+2+3+4+5+6+7+8+9+10 -->55
s=int(input())
e=int(input())
s=0
for i in range(s,e+1):
    s=s+i#s=0+1-->1
         #s=1+2 -->3
         #s=3+3 -->6
#           s=6+4-->10
#           s=10+5 -->15
#           s=15+6 -->21
#           s=21+7 -->28
#           s=28+8 -->36
#           s=36+9 --->45
#           s=45+10 -->55
print(s)
d=0
while(s>0):
    l=s%10
    d=d+l
    s=s//10
print(d)
```

```
1
50
1275
15
```

Tasks

- Find the Factors of Given Number range is 1 to 1000
- find divisibilty factors of 100 and 11 range is 1 to 1000
- input:1234567865 hint :value is 10 digit number output:valid or invalid
- input:9876543251 hint :9*10+89+78+67+56+45+34+23+52+1*1 output:339%11

```
valid or invalid
```

In [6]:

```
import keyword

print(keyword.kwlist)


```

```
['False', 'None', 'True', 'and', 'as', 'assert', 'async', 'await', 'break', 'cl
ass', 'continue', 'def', 'del', 'elif', 'else', 'except', 'finally', 'for', 'fr
om', 'global', 'if', 'import', 'in', 'is', 'lambda', 'nonlocal', 'not', 'or',
'pass', 'raise', 'return', 'try', 'while', 'with', 'yield']
```

Varible Declaration

In [7]:

```
d=20 #Int
# int d=20
```

In [8]:

```
d
```

Out[8]:

```
20
```

In [9]:

```
a,b,c=20,30,10
```

In [10]:

```
c+a
```

Out[10]:

```
30
```

In [11]:

```
c-b
```

Out[11]:

```
-20
```

In [13]:

```
print("Hello World")
```

```
Hello World
```

In [22]:

```
a=43
b=126
type(a)

```

Out[22]:

```
int
```

In [15]:

```
a+b
```

Out[15]:

```
169
```

In [16]:

```
print("the Addition of a and b is ",a+b)
```

```
the Addition of a and b is  169
```

In [17]:

```
print("the Subtration  of a and b is ",a-b)
```

```
the Subtration  of a and b is  -83
```

In [18]:

```
print("The Multiplication of ",a, "and",b,"is ",a*b)
```

```
The Multiplication of  43 and 126 is  5418
```

In [20]:

```
#Greetings to your friend
greet=input("Enter the Geetings..:")
friendName=input("Enter the Friend Name :")
print("Hi Hello ",friendName,greet)

```

```
Enter the Geetings..:good evening
Enter the Friend Name :sai
Hi Hello  sai good evening
```

In [24]:

```
x=int(input("Enter the First Number"))
y=int(input("Enter the Second Number"))
print("Addition Of x and y is ",x+y)
```

```
Enter the First Number3
Enter the Second Number4
Addition Of x and y is  7
```

In [23]:

```
type(x)
```

Out[23]:

```
str
```

In [ ]:

```
#String:
It is a combination set of charecters ,digits,alphanumaric,special char
->""  or ''
s="hello6636sjhfjksdfkjh*%&&%&^%"
short form is str
```

In [25]:

```
s='apssdc'

```

In [27]:

```
s[5]
```

Out[27]:

```
'c'
```

In [44]:

```
add()
```

```
5
```

In [ ]:

```
#Split
#Join

```

In [28]:

```

#Split()
ns="Hello apssdc cbit"
ns.split()

```

Out[28]:

```
['Hello', 'apssdc', 'cbit']
```

In [29]:

```
ns.split(",")
```

Out[29]:

```
['Hello apssdc cbit']
```

In [30]:

```

s1="python@program@cbit@apssdc@ece"
s1.split("@")
```

Out[30]:

```
['python', 'program', 'cbit', 'apssdc', 'ece']
```

In [33]:

```
#Join()
s2="hello"
s3="#".join(s2)
s3
```

Out[33]:

```
'h#e#l#l#o'
```

In [34]:

```
s3.split("#")
```

Out[34]:

```
['h', 'e', 'l', 'l', 'o']
```

In [35]:

```
s1.split("@")
```

Out[35]:

```
['python', 'program', 'cbit', 'apssdc', 'ece']
```

In [37]:

```
c="aps-sdc-apassembl-yarthae-mtic"
c.split("-")
```

Out[37]:

```
['aps', 'sdc', 'apassembl', 'yarthae', 'mtic']
```

In [38]:

```
singleLetter="srikanth"
newsrikath="^".join(singleLetter)
newsrikath
```

Out[38]:

```
's^r^i^k^a^n^t^h'
```

In [39]:

```
newsrikath.split("^")
```

Out[39]:

```
['s', 'r', 'i', 'k', 'a', 'n', 't', 'h']
```

In [40]:

```
type(newsrikath)
```

Out[40]:

```
str
```

In [41]:

```
print(dir(str))
```

```
['__add__', '__class__', '__contains__', '__delattr__', '__dir__', '__doc__',
'__eq__', '__format__', '__ge__', '__getattribute__', '__getitem__', '__getnewa
rgs__', '__gt__', '__hash__', '__init__', '__init_subclass__', '__iter__', '__l
e__', '__len__', '__lt__', '__mod__', '__mul__', '__ne__', '__new__', '__reduce
__', '__reduce_ex__', '__repr__', '__rmod__', '__rmul__', '__setattr__', '__siz
eof__', '__str__', '__subclasshook__', 'capitalize', 'casefold', 'center', 'cou
nt', 'encode', 'endswith', 'expandtabs', 'find', 'format', 'format_map', 'inde
x', 'isalnum', 'isalpha', 'isascii', 'isdecimal', 'isdigit', 'isidentifier', 'i
slower', 'isnumeric', 'isprintable', 'isspace', 'istitle', 'isupper', 'join',
'ljust', 'lower', 'lstrip', 'maketrans', 'partition', 'replace', 'rfind', 'rind
ex', 'rjust', 'rpartition', 'rsplit', 'rstrip', 'split', 'splitlines', 'startsw
ith', 'strip', 'swapcase', 'title', 'translate', 'upper', 'zfill']
```

Funtions In Python

- Set of Statemnets called Funtion
    - Group of Instrations called Statement /Program
- Code Reusability

In [ ]:

```
 Two :
        Inbuilt

        User Defiend
            * four typed
                 1.Funtion with arg[] and with return value
                 2.Funtion with arg[] and with out return value
                 3.Funtion with out arg[] and with return value
                 4.Funtion with out arg[] and with out return value
```

In [ ]:

```
#Funtion Syntax
def funtion_name():
    ....stmts
funtion_name()
============
#Funtion Syntax
def funtion_name(a,b):
    ....stmts
funtion_name(a,b)
=============
#Funtion Syntax
def funtion_name():
     return stmts
funtion_name()
==============
#Funtion Syntax
def funtion_name(a,b):
     return stmts
funtion_name(5,6)
```

In [45]:

```
#Example of
def add():
    print(a+b)
a=2
b=3
add()
```

```
5
```

In [46]:

```
def add(a,b):
    print(a+b)
a=2
b=3
add(a,b)
```

```
5
```

In [53]:

```
def add(a,b):
    return a+b
add(7,9)
```

Out[53]:

```
16
```

In [54]:

```
def add():
    return 7+5
add()
```

Out[54]:

```
12
```

In [ ]:

```
#Calculator application
User Choice is 1.Addition 2.Subtration 3. Multiplication 4.Floor Division 5.

```

In [2]:

```
#Definning Addtion
def addition(a,b):
    print("Addition ",a+b)
addition(2,3)
```

```
Addition  5
```

In [3]:

```
#Defining Addtion
def subtration(a,b):
    print("subtration ",a-b)
subtration(2,3)
```

```
subtration  -1
```

In [4]:

```
#Defining Addtion
def Multiplicstion(a,b):
    print("Multiplication ",a*b)
Multiplicstion(2,3)
```

```
Multiplication  6
```

In [5]:

```
#Defining Addtion
def div(a,b):
    print("floor div is ",a//b)
div(5,2)
```

```
floor div is  2
```

In [ ]:

```

```

In [ ]:

```python
def CalculatorApp():
    while True:
        print("1.Addition \n2.Subtration \n3. Multiplication \n4.Floor Divis
        uc=int(input("Enter the User choice: "))
        if uc==1:
            addition(3,1)
            break
        elif uc==2:
            subtration(8,4)
        elif uc==3:
            Multiplicstion(3,3)
        elif uc==4:
            div(8,2)
        elif uc==5:
            return False
        continue

CalculatorApp()
```

In [17]:

```python
# funtion with arguments and with return value
# is prime or not
def isprime(n):
    c=0
    for i in range(1,n+1):
        if(n%i==0):
            c=c+1
    if(c==2):
        return n
```

In [15]:

```python
# 10-->2 3 5 7
v=int(input("Enter Range :"))
for i in range(2,v+1):
    isprime(i)

```

```
Enter Range :10
2
3
5
7
```

In [16]:

```
lb=int(input())
ub=int(input())
for i in range(lb,ub+1):
    isprime(i)
```

```
10
20
11
13
17
19
```

In [22]:

```
# input:10
# ouput:17
n=int(input())
s=0
for i in range(2,n+1):
    if(isprime(i)):
        s=s+isprime(i)
print(s)
```

```
100
1060
```

In [23]:

```
lb=int(input())
ub=int(input())
s=0
for i in range(lb,ub+1):
    if(isprime(i)):
        s=s+isprime(i)
print(s)

```

```
10
20
60
```

In [28]:

```
# Leap Year
def isleapyear(n):
    if(n%400==0 or (n%100!=0 and n%4==0)):
        return n
isleapyear(655)
```

In [31]:

```
lb=int(input())
ub=int(input())
c=0
for i in range(lb,ub+1):
    if(isleapyear(i)):
        c=c+1
        print(c,"-->",i)
# print(c)
```

```
2000
2020
1 --> 2000
2 --> 2004
3 --> 2008
4 --> 2012
5 --> 2016
6 --> 2020
```

In [2]:

```
# 6 --->1+2+3 -->6
# perfect Number
def isperfect(n):
    s=0
    for i in range(1,n):
        if(n%i==0):
            s=s+i
    if(n==s):
        return s
isperfect(28)
```

Out[2]:

```
28
```

In [3]:

```
lb=int(input())
ub=int(input())
for i in range(lb,ub+1):
    if(isperfect(i)):
        print(i)
```

```
1
30
6
28
```

In [4]:

```
print(dir(str))
```

```
['__add__', '__class__', '__contains__', '__delattr__', '__dir__', '__doc__',
'__eq__', '__format__', '__ge__', '__getattribute__', '__getitem__', '__getnewa
rgs__', '__gt__', '__hash__', '__init__', '__init_subclass__', '__iter__', '__l
e__', '__len__', '__lt__', '__mod__', '__mul__', '__ne__', '__new__', '__reduce
__', '__reduce_ex__', '__repr__', '__rmod__', '__rmul__', '__setattr__', '__siz
eof__', '__str__', '__subclasshook__', 'capitalize', 'casefold', 'center', 'cou
nt', 'encode', 'endswith', 'expandtabs', 'find', 'format', 'format_map', 'inde
x', 'isalnum', 'isalpha', 'isascii', 'isdecimal', 'isdigit', 'isidentifier', 'i
slower', 'isnumeric', 'isprintable', 'isspace', 'istitle', 'isupper', 'join',
'ljust', 'lower', 'lstrip', 'maketrans', 'partition', 'replace', 'rfind', 'rind
ex', 'rjust', 'rpartition', 'rsplit', 'rstrip', 'split', 'splitlines', 'startsw
ith', 'strip', 'swapcase', 'title', 'translate', 'upper', 'zfill']
```

In [5]:

```
s=" cbit "
```

In [6]:

```
s
```

Out[6]:

```
' cbit '
```

In [7]:

```
s.strip()
```

Out[7]:

```
'cbit'
```

In [8]:

```
s.rstrip()
```

Out[8]:

```
' cbit'
```

In [9]:

```
s.lstrip()
```

Out[9]:

```
'cbit '
```

In [10]:

```
s.islower()
```

Out[10]:

```
True
```

In [11]:

```
s.isupper()
```

Out[11]:

```
False
```

In [12]:

```
s1="80340ehjhfi8934^^%^$$%r8903jehfRYRYiuehfio"
s1.isalnum()

```

Out[12]:

```
False
```

In [21]:

```python
s1.count('j')
```

Out[21]:

```
2
```

In [16]:

```python
for i in s1:
    if i.isdigit():
        print(i,end=" ")



```

```
8 0 3 4 0 8 9 3 4 8 9 0 3
```

In [ ]:

```python
ns="banana 60 apple 20 graps 40 "
output:120

```

In [17]:

```python
#input:vijay10022@gmail.com
#output:character count=13
#          digit count = 5
#          special count =2
def fun(data):
    c=0
    d=0
    s=0
    for i in data:
        if(i.isdigit()):
            d=d+1
        elif(i.isalpha()):
            c=c+1
        else:
            s=s+1
    print("charater Count:",c)
    print("Digit Count :",d)
    print("Sepecial Count :",s)
fun("vijay10022@gmail.com")
```

```
charater Count: 13
Digit Count : 5
Sepecial Count : 2
```

In [22]:

```python
ord('a')# convert charaer to Ascii

```

Out[22]:

```
97
```

In [23]:

```python
chr(97)# convertion of ascii to charater
```

Out[23]:

```
'a'
```

In [24]:

```
for i in range(0,129):
    print(i,"-->",chr(i))
```

```
50 --> 2
51 --> 3

52 --> 4
53 --> 5
54 --> 6
55 --> 7
56 --> 8
57 --> 9
58 --> :
59 --> ;
60 --> <
61 --> =
62 --> >
63 --> ?
64 --> @
65 --> A
66 --> B
67 --> C
68 --> D
69 --> E
```

In [35]:

```
data="vijayKUMAR10022@gmail.com"
def fun2(data):
    c=0
    d=0
    s=0
    for i in range(len(data)):
        #print(ord(data[i]),end=" ")
        if((ord(data[i])>=65 and ord(data[i])<=90) or ((ord(data[i]))>=97 and
                      ord(data[i])<=122)):

            #print(data[i],end=" ")
            c=c+1
        elif ((ord(data[i])>=48) and
              (ord(data[i])<=57)):

             d=d+1
        else:
             s=s+1
    print(c)
    print(d)
    print(s)
fun2(data)
```

```
18
5
2
```

Python Data Stracture

List -->[]

- Collection Hetrogenoues Data set
    - Example digits,chr,spl char,space...etc
- Reprasents symbol like this []
- List Contain Dupliate data set
- List Is a Mutable(can Modify,create,update,delete)

Tuple-->()

Set -->{}

In [36]:

```python
li=[1,3,5,2,8,1]
li
```

Out[36]:

```
[1, 3, 5, 2, 8, 1]
```

In [ ]:

```python
#List indexing
#List Slicing

```

In [37]:

```python
len(li)
```

Out[37]:

```
6
```

In [44]:

```python
sorted(li,reverse=True)#Revered order Sorted
```

Out[44]:

```
[8, 5, 3, 2, 1, 1]
```

In [45]:

```python
sorted(li)#Assending order
```

Out[45]:

```
[1, 1, 2, 3, 5, 8]
```

In [42]:

```python
li.sort()

```

In [43]:

```python
li
```

Out[43]:

```
[1, 1, 2, 3, 5, 8]
```

In [46]:

```python
min(li)
```

Out[46]:

```
1
```

In [47]:

```python
max(li)
```

Out[47]:

```
8
```

In [48]:

```
print(dir(list))
```

```
['__add__', '__class__', '__contains__', '__delattr__', '__delitem__', '__dir_
_', '__doc__', '__eq__', '__format__', '__ge__', '__getattribute__', '__getitem
__', '__gt__', '__hash__', '__iadd__', '__imul__', '__init__', '__init_subclass
__', '__iter__', '__le__', '__len__', '__lt__', '__mul__', '__ne__', '__new__',
'__reduce__', '__reduce_ex__', '__repr__', '__reversed__', '__rmul__', '__setat
tr__', '__setitem__', '__sizeof__', '__str__', '__subclasshook__', 'append', 'c
lear', 'copy', 'count', 'extend', 'index', 'insert', 'pop', 'remove', 'revers
e', 'sort']
```

In [49]:

```

li.remove(8)
```

In [50]:

```
li
```

Out[50]:

```
[1, 1, 2, 3, 5]
```

In [51]:

```
li.pop()
```

Out[51]:

```
5
```

In [52]:

```
li.append(10)
```

In [53]:

```
li
```

Out[53]:

```
[1, 1, 2, 3, 10]
```

In [56]:

```
uniq=[]
for i in li:
    if i not in uniq:
        uniq.append(i)
uniq
print("Total is ",sum(uniq))


```

```
Total is  16
```

In [ ]:

```
Task:
names=['srirama','sai','sairam','Veera venkata satyanarayana','venkatesa',"H
```