Pattern odd numbers

```
17
18  rows=10
19  for i in range(rows+1,1,-1):
20      for j in range(1,(i*2)-1):
21          if j%2!=0:
22              print(j,"",end="")
23      print("\n")
```

```
1 3 5 7 9 11 13 15 17 19

1 3 5 7 9 11 13 15 17

1 3 5 7 9 11 13 15

1 3 5 7 9 11 13

1 3 5 7 9 11

1 3 5 7 9

1 3 5 7

1 3 5

1 3

1
```

Number to Words

```
[109]: num={'1':"one",'2':"two",'3':"three",'4':"four",'5':"five",
            '6':"six",'7':"seven",'8':"eight",'9':"nine",'0':"zero"}
       n=input("Enter any number: ")
       for i in n:
           print(i,":",numDect(i))
```

```
Enter any number:  9846
9 : nine
8 : eight
4 : four
6 : six
```

```
[ ]:
```

Math class and their functions(add,sub,mul,div)

```
[135]: class Math:
           def setData(self,a,b):
               self.a=a
               self.b=b
               print("a=",a)
               print("b=",b)
           def add(self):
               print("add =",self.a+self.b)
           def sub(self):
               print("sub =",self.a-self.b)
           def mul(self):
               print("mul =",self.a*self.b)
           def div(self):
               print("div =",self.a/self.b)

       a=Math()
       a.setData(36,12)
       a.add()
       a.sub()
       a.mul()
       a.div()
```

```
a= 36
b= 12
add = 48
sub = 24
mul = 432
div = 3.0
```

Given an array of non-negative integers arr[], the task is to find a pair (n, r) such that nPr is maximum possible and r ≤ n. nPr = n! / (n – r)!

Examples:

Input: arr[] = {5, 2, 3, 4, 1} Output: n = 5 and r = 4

5P4 = 5! / (5 – 4)! = 120 which is maximum possible.

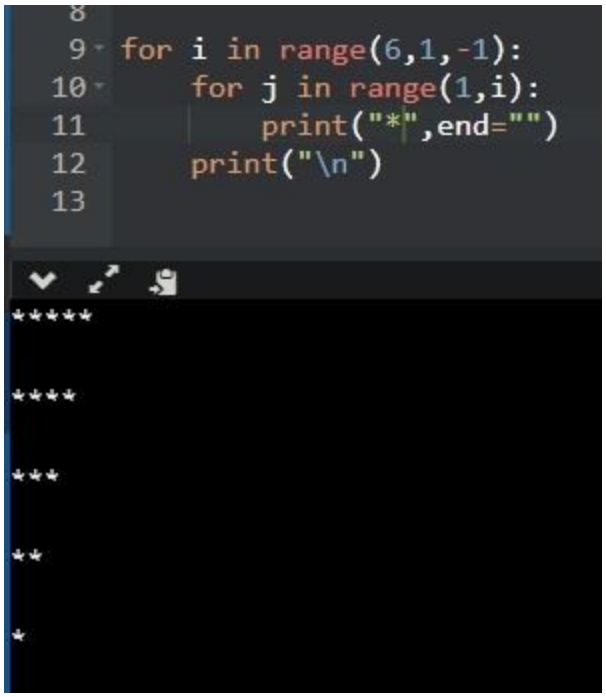
Input: arr[] = {0, 2, 3, 4, 1, 6, 8, 9} Output: n = 9 and r = 8

```
 9  import numpy as np
10  import math
11  ar=np.array([0, 2, 3, 4, 1, 6, 8, 9])
12  print("Array",ar)
13  print("factorial of",max(ar),"is",math.factorial(max(ar)))
14
```

input

```
Array [0 2 3 4 1 6 8 9]
factorial of 9 is 362880
```

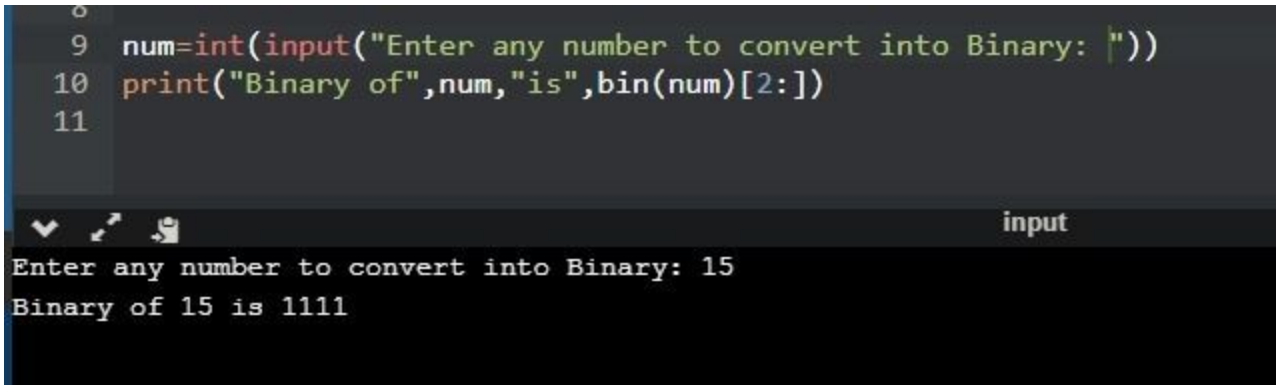
Program to print inverted half pyramid using an asterisk (star)



```
for i in range(6,1,-1):
    for j in range(1,i):
        print("*",end="")
    print("\n")
```

```
*****

****

***

**

*
```

Convert a given number to binary format.



```
num=int(input("Enter any number to convert into Binary: "))
print("Binary of",num,"is",bin(num)[2:])
```

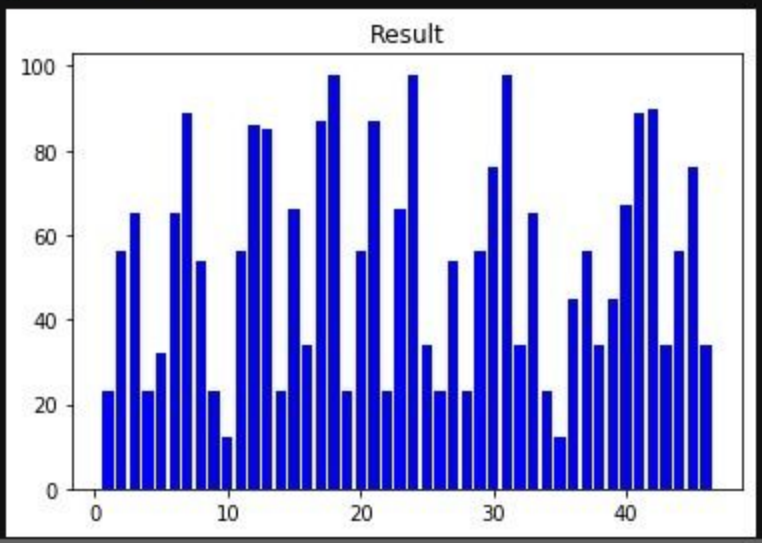
```
Enter any number to convert into Binary: 15
Binary of 15 is 1111
```

Assignment: plot student's score data using matplotlib

[10]:
```
import pyodbc

conn = pyodbc.connect('Driver={Microsoft Access Driver (*.mdb, *.accdb)};\
                       DBQ=d:/students.accdb;')
cursor = conn.cursor()

try:
    sql="select * from vitaStudents"
    cursor.execute(sql)
    result=cursor.fetchall()
    #print(result,type(result))
    for row in result:
        print (row)
except Exception as arg:
    print("Error is:",arg)
```

```
(1, 'ABHED MHATRE', 23)
(2, 'ABHISHEK JAIN', 56)
(3, 'ABHISHEK WALANJ', 65)
(4, 'APURVA WAGHMODE', 23)
(5, 'ASHISH PURALKAR', 32)
(6, 'BHAVESH MAKWANA', 65)
(7, 'BHAVESH SAWANT', 89)
(8, 'CHANDAN BHARDWAJ', 54)
(9, 'DHEERAJKUMAR DEORE', 23)
(10, 'DIPENDU BENDALE', 12)
```

[44]:
```
import matplotlib.pyplot as plt
import numpy as np

res=list(zip(*result))
score=res[2]
prn=res[0]

a=np.arange(1,101)
plt.bar(prn,score,color="b")
plt.xlabel="students"
plt.ylabel="name"
plt.title("Result")
plt.show()
```



Find permutations of a astring

```
[10]: from itertools import permutations

      string="asd"
      for p in permutations(string):
          print(''.join(p))
```

```
asd
ads
sad
sda
das
dsa
```

Given four integers m, n, a, b. Find how many integers from range m to n are divisible by a or b.
Examples : Input: 3 11 2 3 Output: 6
Explanation: m = 3, n = 11, a = 2, b = 3
There are total 6 numbers from 3 to 11 which are divisible by 2 or 3 i.e, 3, 4, 6, 8, 9, 10
Input: arr[] = {11, 1000000, 6, 35} Output: 190475

```
[3]: m=int(input("Enter m : "))
     n=int(input("Enter n : "))
     a=int(input("Enter a : "))
     b=int(input("Enter b : "))

     print("m =",m,"n =",n,"a =",a,"b =",b)
     count=0
     for i in range(m,n+1):
         if i%a==0 or i%b==0:
             count+=1
             #print(i)

     print("There are total",count,"between",m,"and",n,"that are divisible by",a,"or",b)
```

```
Enter m :  11
Enter n :  1000000
Enter a :  6
Enter b :  35
m = 11 n = 1000000 a = 6 b = 35
There are total 190475 between 11 and 1000000 that are divisible by 6 or 35
```

[2]:
```
m=int(input("Enter m : "))
n=int(input("Enter n : "))
a=int(input("Enter a : "))
b=int(input("Enter b : "))

print("m =",m,"n =",n,"a =",a,"b =",b)
count=0
for i in range(m,n+1):
    if i%a==0 or i%b==0:
        count+=1
        print(i)

print("There are total",count,"between",m,"and",n,"that are divisible by",a,"or",b)
```

```
Enter m :  3
Enter n :  11
Enter a :  2
Enter b :  3
m = 3 n = 11 a = 2 b = 3
3
4
6
8
9
10
There are total 6 between 3 and 11 that are divisible by 2 or 3
```

convert hours and minutes in seconds

```
# convert hours and minutes in seconds

hour=3
mins=1
print(hour,"hour to seconds=",hour*60*60,"\n",mins,"mins to seconds=",mins*60)
```

```
3 hour to seconds= 10800
 1 mins to seconds= 60
```