Q.nO1 Number game between user and computer. The user starts by entering either 1 or 2 or 3 digits starting from 1 sequentially. The computer can return either 1 or 2 or 3 next digits in sequence, starting from the max number played by the user. User enters the next 1 or 2 or 3 next digits in sequence, starting from the max number played by the computer. Whoever reaches 20 first wins the game.

```
num_list1=[]
numlist2=[]
num_list3=[]
num-List4=[]
for i in range(1,5):
   if(i==int(input("enter the value")) and i<3):
   continue
   num_list1.append(i)
   print(num_list1)
for j in range(6,10):
    if(j==int(input("enter the value")) and j<8):
   continue
   num_list2.append(j)
   print(num_list2)
for k in range(6,10):
    if(k==int(input("enter the value")) and k<13):
   continue
   num_list3.append(k)
   print(nu_list3)
for l in range(26,20):
    if(l==int(input("enter the value")) and l<18):
   continue
   num_list4.append(l)
```

```
   print(num_list4)
print("system wins the game ")
break:
print("user wins the game ")
```

Q.NO2 Develop a function called ncr(n,r) which computes r-combinations of n-distinct object . use this function to print pascal triangle, where number of rows is the input

```
from math import factorial
rows=int(input("enter the number of rows:  "))
for n in range(rows):
   for space in range(1,rows-n):
      print(end= " ")
for r in range(n+1):
   ncr=factorial(n)//(factorial(r)*factorial(n-r))
   print(ncr, end=" ")
   print(" ")
```

Output:

```
Enter the number of rows:5
          1
        1   1
      1  2   1
    1 3   3   1
  1  4  6   4  1
```

Question 3:

Read a list of n numbers during runtime. Write a Python program to print the repeated elements with frequency count in a list.

```
L1=list()
n=int(input("enter the size of list"))
for a in range(n):
   p=int(input("enter an element"))
   L1.append(p)
print(L1)
num=int(input("enter the value  to be searched: "))
count=0
for ele in L1:
   if(ele==num):
      count=count+1
print(num,"has occured", count, "times:")
```

Out put:

enter the size of list 5

enter an element 2

enter an element 3

enter an element 4

enter an element 5

enter an element 5

[2, 3, 4, 5, 5]

enter the value to be searched: 5

5 has occured 2 times:

Question 4:-

Develop a python code to read matric A of order 2X2 and Matrix B of order 2X2 from a file and perform the addition of Matrices A & B and Print the results.

```
f=open("demo.txt","w")
f.write("12345678")
f.close()
f=open("demo.txt","r")
data_list=f.readlines()
print(data_list)
M1Row1_list=[]
M1Row2_list=[]
M2Row1_list=[]
M2Row2_list=[]
for i in data_list:
    M1Row1_list.append(int(i[0]))
    M1Row1_list.append(int(i[1]))
    M1Row2_list.append(int(i[2]))
    M1Row2_list.append(int(i[3]))
    M2Row1_list.append(int(i[4]))
    M2Row1_list.append(int(i[5]))
    M2Row2_list.append(int(i[6]))
    M2Row2_list.append(int(i[7]))
    print(M1Row1_list)
    print(M1Row2_list)
    print(M2Row1_list)
    print(M2Row2_list)
    sumrow1_list=[]
    for r in range(len(M1Row1_list)):
        sumrow1_list.append(M1Row1_list[r]+M2Row1_list[r])
        print(sumrow1_list)
```

```
    sumrow2_list=[]
  for j in range(len(M2Row1_list)):
    sumrow2_list.append(M1Row2_list[r]+M2Row2_list[r])
    print(sumrow2_list)
```

Output:

['12345678']

[1, 2]

[3, 4]

[5, 6]

[7, 8]

[6]

[6, 8]

[10]

[10, 12]

Question 5

Write a program that overloads the + operator so that it can add two objects of the class Fraction.

Fraction can be considered of the for P/Q where P is the numerator and Q is the denominator

```
def GCD(num, deno):
if(deno == 0):
return num
else:
return GCD(deno,num%deno)
class Fraction:
```

```
def __init__(self):
self.num = 0
self.deno = 1
def get(self):
self.num  =  int(input("enter the numerator : "))
self.deno = int(input("enter the denominator : "))
def simplify(self):
common_divisor = GCD(self.num,self.deno)
self.num //= common_divisor
self.deno //= common_divisor
def __add__(self,F)
Temp = Fraction()
Temp.num = (self.num*F.deno)+(F.num*self.deno)
Temp.deno = self.deno * F.deno
return Temp
def display(self):
self.simplify()
print(self.num,"/",self.deno)
F1= Fraction()
F1.get()
F2=Fraction()
F2.get()
F3=fraction()
F3=F1+F2
print("resultant fraction is : ")
F3.display()
```

Output:

enter the numerator : 4

enter the denominator : 10

enter the Numerator : 2

enter the denominator : 5

resultant fraction is **: 4 / 5**