In [5]:
```
import random
import matplotlib.pyplot as plt
%matplotlib inline

Q = 1
N = 10**4
Arr = []
for i in range (N):
    Arr += [random.expovariate(Q)]


def fact(n):
    if n == 0:
        return 1
    f = 1
    for i in range(1, n + 1):
        f *= i
    return f

def countFirst(a, n, q, k, Y):
    sample = a[:n]
    return abs((fact(k)/ countKNorm(a,n,q,k,Y))**(1/k) - q)


def countKNorm(a, n, q, k, Y):
    if n == 1:
        return a[0]**k
    else:
        return Y[n - 2]*(n-1) +a[n-1]**k



def plot(countFunc, arr, n, q, k, col, name):
    x = range(n)
    Y = []
    for i in range(n):
         Y += [countFunc(arr, i + 1, q, k, Y) / (n+1)]
    plt.plot(x, Y, color = col, label = name)

def research1(q):
    plot(countFirst, Arr, N, q, 1, 'r', "1")
    plot(countFirst, Arr, N, q, 2, 'g', "2")
    plot(countFirst, Arr, N, q, 3, 'b', "3")
    plot(countFirst, Arr, N, q, 4, 'y', "4")
    plot(countFirst, Arr, N, q, 5, 'm', "5")

    legend = plt.legend(loc='upper center', shadow=True, fontsize='x-large')
    legend.get_frame().set_facecolor('#00FFCC')

    plt.show()

def research2(q):

    plot(countFirst, Arr, N, q, 6, 'r', "6")
    plot(countFirst, Arr, N, q, 7, 'g', "7")
    plot(countFirst, Arr, N, q, 8, 'b', "8")
    plot(countFirst, Arr, N, q, 10, 'y', "10")
    plot(countFirst, Arr, N, q, 15, 'm', "35")


    legend = plt.legend(loc='upper center', shadow=True, fontsize='x-large')
    legend.get_frame().set_facecolor('#00FFCC')
```