In [2]:

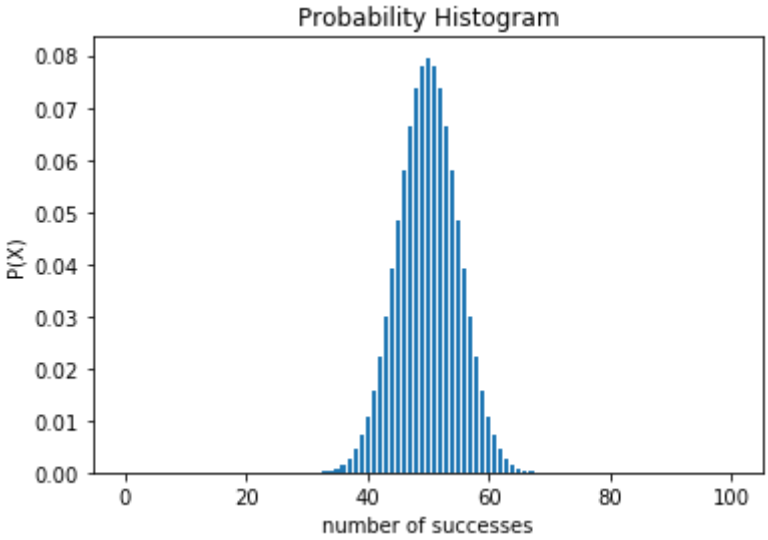
```
import math
import matplotlib.pyplot as plt
import numpy as np
from math import factorial as fact

# nCk = n! / k! (n - k)!
def combination(n,x):
    comb = fact(n)/(fact(x) * fact(n-x))
    return comb

# bin(n, p) = nCk * pow(p, k) * pow((1-p), n - k)
# k ranges from 0 to n => total values = n + 1
def binom(n,p):
    pmf = []
    x = np.arange(n + 1)
    for k in range (0,n+1):
        prob = combination(n,k)*math.pow(p,k)*math.pow(1-p,n-k)
        pmf.append(prob)

    plt.bar(x, pmf)
    plt.xlabel('number of successes')
    plt.ylabel('P(X)')
    plt.title('Probability Histogram')
#    plt.xticks(x + 0.5,x)
    plt.show()

binom(100,0.5)
```



In [3]:

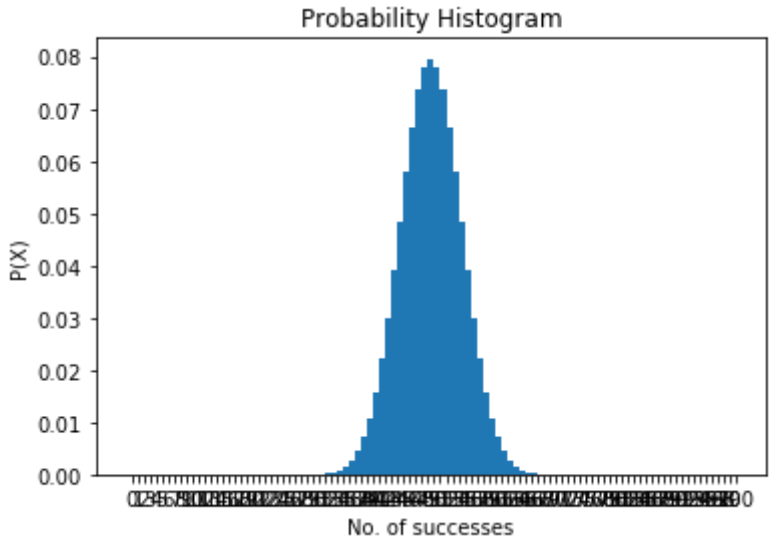
```
#Binomial Distribution using scipy.stats.binom.pmf


import matplotlib.pyplot as plt

import scipy.stats
import numpy as np
p = []
n = 100
x = np.arange(n+1)
for i in range(0, n + 1):
    #pmf(x, n, p) returns the probability value for this set of (x, n, p)
    ax = scipy.stats.binom.pmf(i, n, 0.5)
    p.append(ax)

plt.bar(x, p, width = 1)
```

```
plt.xlabel('No. of successes')
plt.ylabel('P(X)')
plt.xticks(x + 0.5, x)
plt.title('Probability Histogram')
plt.show()
```



In [ ]:

In [ ]: