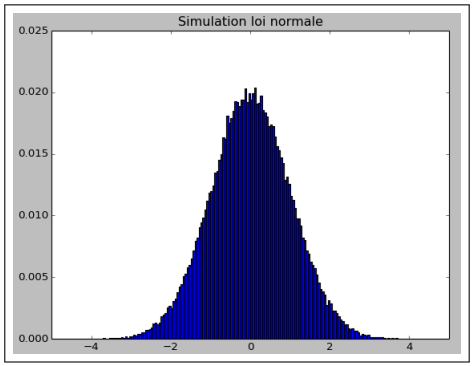
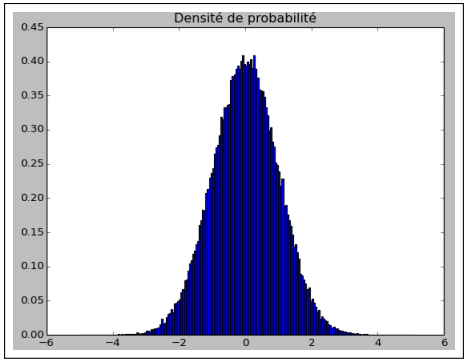
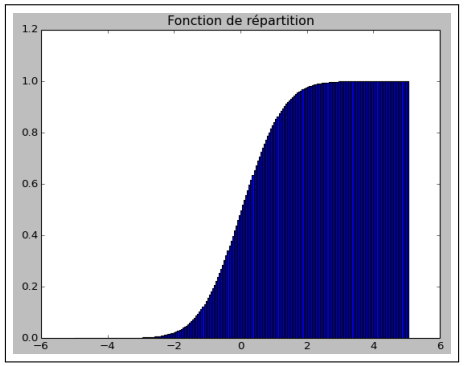
```
from random import normalvariate
from math import floor # Fonction partie entière

def varNormale(mu,sigma):
    return normalvariate(mu,sigma)

# 1)
Max = 5

def simulNormale(N,n,mu,sigma):
    FreqX = [0] * (2*n+1)
    Nbre = 0
    for k in range(N):
        x = varNormale(mu,sigma)
        if -Max <= x <= Max:
            i = floor(n*x/Max) + n
            FreqX[i] += 1
            Nbre += 1
    return [x/Nbre for x in FreqX]

# 2)
N = 100000
n = 100
I = np.linspace(-Max,Max,2*n+1)
FreqX = simulNormale(N, n, 0, 1)
plt.figure(1)
plt.clf()
plt.title("Simulation loi normale")
plt.bar(I,FreqX,width = Max/n)
plt.show()

# 3)
fx = [0] * (2*n+1)
for k in range(2*n+1):
    fx[k] = FreqX[k] * (2*n+1)/(2*Max)
plt.figure(2)
plt.clf()
plt.title("Densité de probabilité")
plt.bar(I,fx,width=Max/n)
plt.show()
```

```
# 4)
Fx = [0] * (2*n+1)
for k in range(2*n+1):
    for i in range(0,k):
        Fx[k] += FreqX[i]
plt.figure(3)
plt.clf()
plt.title("Fonction de répartition")
plt.bar(I,Fx,width=Max/n)
plt.show()
```

Ce code produit les graphiques suivants :






Fonction de répartition
1.2
1.0
0.8
0.6
0.4
0.2
0.0
−6
−4
−2
0
2
4
6