Simulations en classe de première

Représentation de la moyenne des valeurs des 100 échantillons

```
1 from math import*
2 from numpy import*
 3 from random import*
  4 from matplotlib import*
  5 from matplotlib.pyplot import*
  7 def moyenne(n):
. 8
       L1=[]
. 9
        for j in range(n):
10
           P=random()
11
           if P<=0.6:
12
               P=1000
13
           elif P<=0.8:
14
               P=1050
15
           else:
16
               P=1100
17
           L1.append(P)
18
       return(mean(L1))
 19
 20 def graph(n,E):
21
       for i in range(E):
           plot(i,moyenne(n),'+',color='red')
23
       axis([0,100,1000,1100])
24
        plot([0,100],[1022,1022],color='green')
25
        plot([0,100],[1038,1038],color='green')
26
       show()
· 27 graph(50,100)
 28
29
```

Proportion des échantillons tels que la moyenne m vérifie $\mu - \frac{2\sigma}{\sqrt{n}} \le m \le \mu + \frac{2\sigma}{\sqrt{n}}$

```
1 from math import*
                                          20 def écart(n,N):
 2 from numpy import*
                                                 Compt=0
 3 from random import*
                                                  for i in range(N):
 4 from matplotlib import*
                                          23
                                                      if abs(moyenne(n)-1030)<=8:</pre>
 5 from matplotlib.pyplot import*
                                          24
                                                          Compt=Compt+1
 6
                                          25
                                                  return Compt/N
 7 def moyenne(n):
                                          26
 8
       L1=[]
                                          27 def graphe(n,N,E):
 9
       for j in range(n):
                                          28
                                                  L=[]
            P=random()
10
                                          29
                                                  for k in range(E):
            if P<=0.6:
11
                                          30
                                                      L.append(écart(n,N))
12
                P=1000
                                          31
                                                  axis([0,E,0.0,1])
            elif P<=0.8:
13
                                                  xlabel("Nombre d'échantilons")
                                          32
14
                P=1050
                                                  ylabel("Proportions observées en bleu")
                                          33
15
            else:
                                          34
                                                 title("Programme première")
16
                P=1100
                                          35
                                                  grid(True)
17
            L1.append(P)
                                          36
                                                 plot(L, 'b-')
18
       return(mean(L1))
                                          37
                                                  show()
19
                                          38 graphe(100,100,100)
```