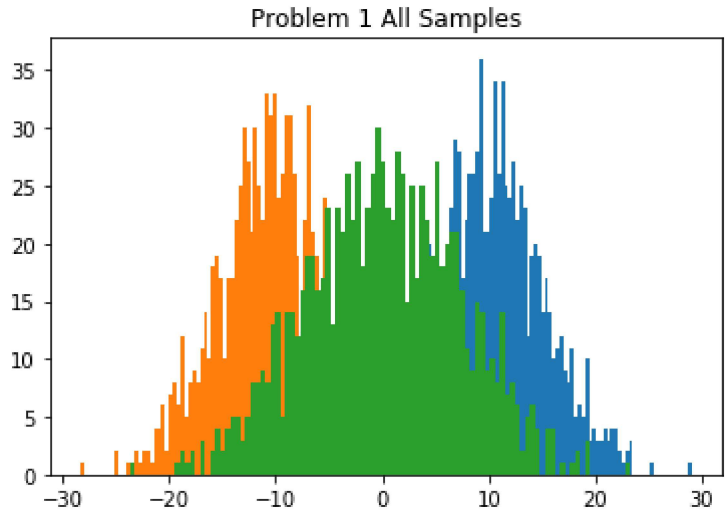
Andriy Dashko - ard2682
Justin Hoang - jah7399

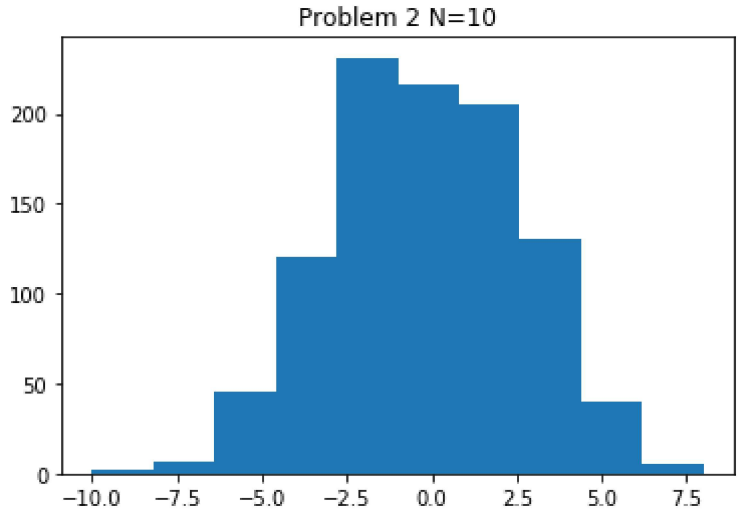
In [2]:
```
import numpy as np
import scipy
from scipy import signal
import matplotlib.pyplot as plt
%matplotlib inline
import warnings
warnings.filterwarnings('ignore')
import pandas as pd
from sklearn.decomposition import PCA
from sklearn.feature_selection import SelectKBest
from sklearn.feature_selection import mutual_info_classif
```

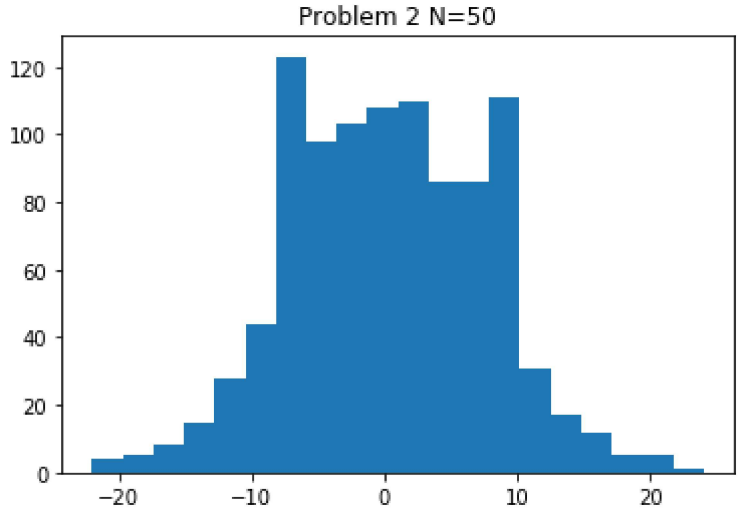
In [3]:
```
#Problem #1
mu = 10
sigma = 5

gaussSample = scipy.random.normal(mu,sigma,1000)
gaussSample2 = scipy.random.normal(-mu,sigma,1000)
sumSample = np.add(gaussSample, gaussSample2)

plt.hist(gaussSample, 100)
plt.hist(gaussSample2, 100)
plt.hist(sumSample, 100)
plt.title("Problem 1 All Samples")
plt.show()

print("Mean:", sumSample.mean())
print("Variance:", sumSample.std()**2)
```

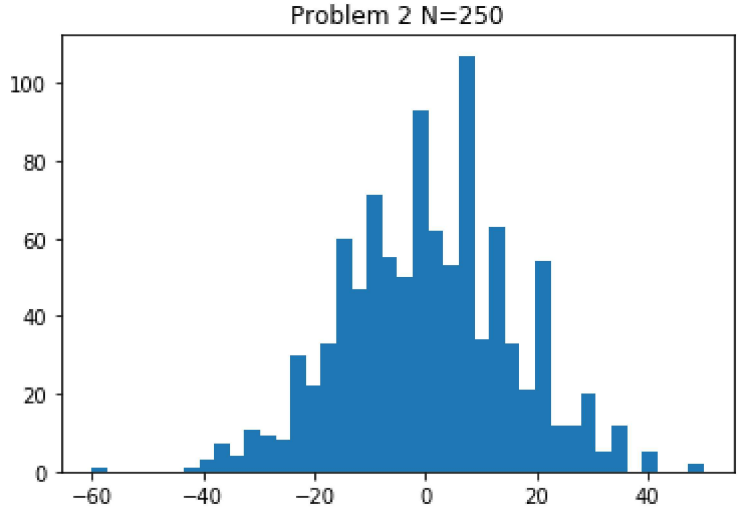



```
Mean: 0.110525769394
Variance: 52.425856078
```

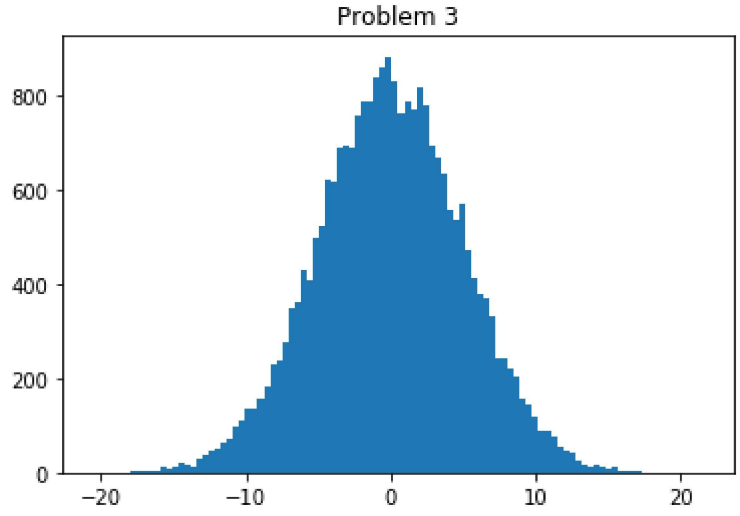
In [4]:
```
#Problem #2
T10 = [0] * 1000
T50 = [0] * 1000
T250 = [0] * 1000

for i in range(0,1000):
    Z10 = np.random.randint(2, size=10)

    for j in range(0,10):
        if Z10[j] == 0:
            Z10[j] = -1
   # print(Z10)
    Z10n = np.sum(Z10)

    T10[i] = Z10n

    Z50 = np.random.randint(2, size=50)

    for j in range(0,50):
        if Z50[j] == 0:
            Z50[j] = -1
    Z50n = np.sum(Z50)

    T50[i] = Z50n

    Z250 = np.random.randint(2, size=250)

    for j in range(0,250):
        if Z250[j] == 0:
            Z250[j] = -1
    Z250n = np.sum(Z250)

    T250[i] = Z250n

plt.hist(T10, 10)
plt.title("Problem 2 N=10")
plt.show()
plt.hist(T50, 20)
plt.title("Problem 2 N=50")
plt.show()
plt.hist(T250, 40)
plt.title("Problem 2 N=250")
plt.show()
```


Problem 2 N=10
200
150
100
50
0
−10.0
−7.5
−5.0
−2.5
0.0
2.5
5.0
7.5


Problem 2 N=50
120
100
80
60
40
20
0
−20
−10
0
10
20

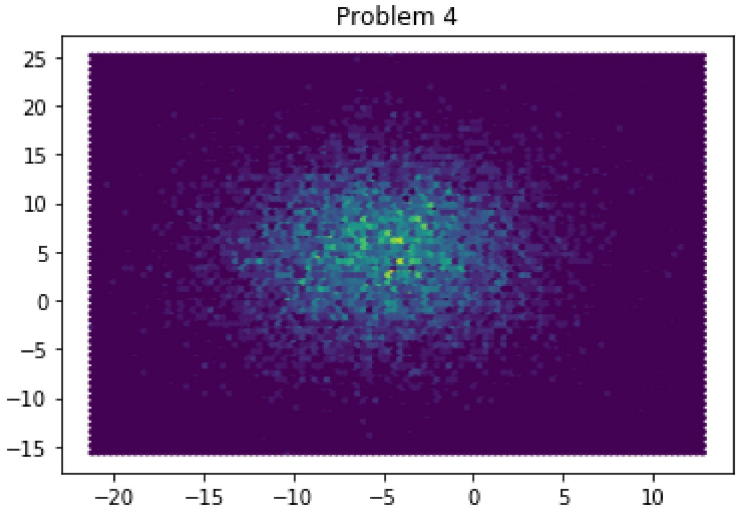

Problem 2 N=250
100
80
60
40
20
0
−60
−40
−20
0
20
40

In [5]:
```
#Problem #3
mu = 0
sigma = 5

gaussSample = scipy.random.normal(mu,sigma,25000)
plt.hist(gaussSample, 100)
plt.title("Problem 3")
plt.show()

mean = np.sum(gaussSample)/25000
print("Mean:", mean)
variance = np.sum((gaussSample - mean)**2)/25000
print("Variance:", variance)
```




```
Mean: -0.00957068786663
Variance: 25.1782269498
```

In [13]:
```
#Problem #4
mean = [-5,5]
covariance =[[20, .8],
             [.8, 30]]

A = np.random.multivariate_normal(mean, covariance, 10000)
x,y = A.T

plt.hexbin(x,y)
plt.title("Problem 4")
plt.show()

Mean1 = np.sum(x)/10000
Mean2 = np.sum(y)/10000
meanR = [Mean1, Mean2]

Variance1 = np.sum((x - Mean1)**2)/10000
Variance2 = np.sum((y - Mean2)**2)/10000
Covariance = np.sum((x - Mean1)*(y - Mean2))/10000

meanR = [[Variance1,Covariance],[Covariance,Variance2]]

print("Mean:", meanR)
```



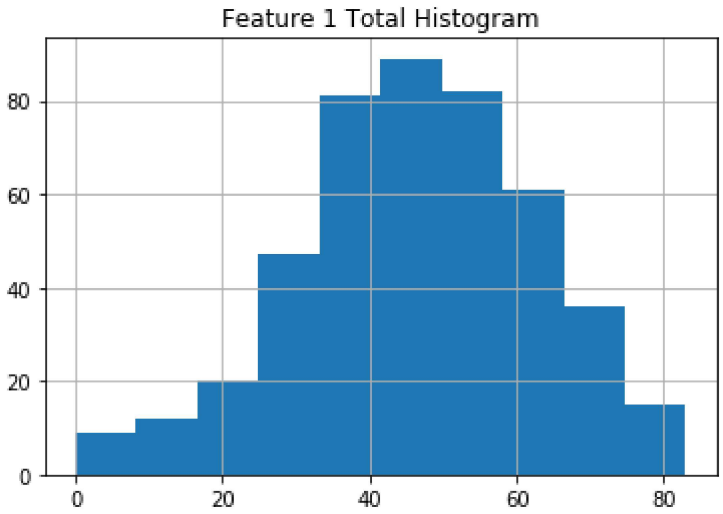

```
Mean: [[19.789723868350922, 0.79843335259521164], [0.79843335259521164, 29.49
3302000476746]]
```

In [16]:
```
#Problem 5
df = pd.read_csv('PatientData.csv', header=None, na_values='?')
rows = df.shape[0]
columns = df.shape[1]
#Part a
print("Part a:")
print("Each row corresponds to one patient, so with,", rows,"rows there are",r
ows, "patients.")
print("Each column corresponds to one feature, so there are", columns, "column
s and features.")
print("\n")
#Part b
print("Part b:")
histAge = df[0].hist()
plt.title("Feature 1 Total Histogram")
plt.show()
print("Feature 1 spans from 0-83 and is fairly normally distributed, so it mos
t likely is Age")
histGender = df[1].hist()
plt.title("Feature 2 Total Histogram")
plt.show()
print("Feature 2 is a binary variable with fairly equal probability for either
 choice, so it most likely is Gender")
df1 = df.sort_values(0)
dfL12 = df1[df1[0] < 12]
dfL12M = dfL12[dfL12[2] < 200]
dfG12 = df1[df1[0] >= 12]
histL12 = dfL12[2].hist(bins=50)
histG12 = dfG12[2].hist(bins=50)
plt.title("Feature 3 Histograms split by Age 12")
plt.show()
print("Average Trygliceride levels Ages <12:(excluding two outliers)", dfL12M.
mean()[2])
print("Average Trygliceride levels Ages >=12:", dfG12.mean()[2])
print("The two points at 608 and 780 for Ages <12 can be explained by the pati
ents not fasting when the measurement was taken because triglyceride levels ca
n spike to 5-10 times the normal level when fasting")
print("Feature 3 averages around 124 for children and then 164 for teenagers a
nd adults. This lines up with trygliceride levels where <150 mg/dL is expected
 for children and adults with borderline tryglicerides have around 150-200 mg/
dL. Therefore, feature 3 is likely trygliceride levels in mg/dL")

df2 = df.sort_values(1)
dfM = df2[df2[1] == 0]
dfF = df2[df2[1] == 1]
histM = dfM[3].hist(bins=50)
histF = dfF[3].hist(bins=50)
plt.title("Feature 4 Histograms split by Gender")
plt.show()
print("Male Average Weight(kg):", dfM.mean()[3])
print("Female Average Weight(kg):", dfF.mean()[3])
print("Feature 4 when split into two data sets based on feature 1 shows that t
here's two distinct means, corresponding to the different genders. Those means
 also happen to be reasonably close to average weight in males and females, an
d the distributions are relatively normal. All things considered, feature 4 is
 most likely weight(kg).")
```

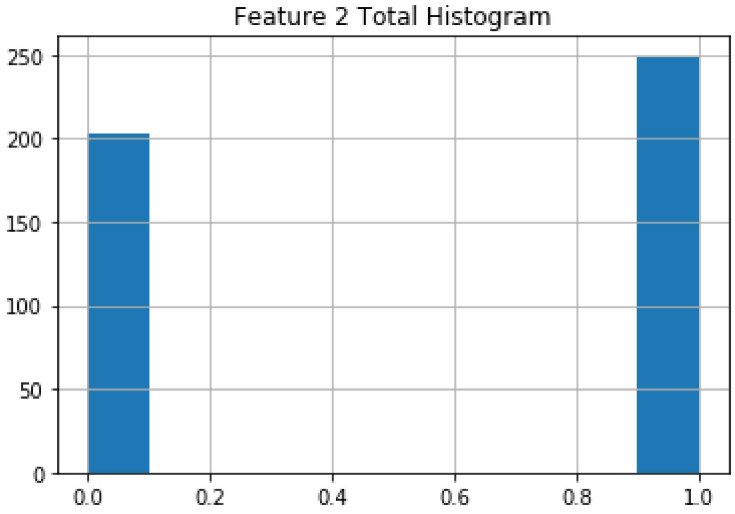
```
print("\n")

#Part c
df.fillna(df.mean(), inplace=True)

#Part d
print("Part d:")
print("The way to test which features are strongly influence the patient condi
tion is using multi-class classification techniques to score the features. Spe
cifically to find the 3 most important features, we're using the Mutual Inform
ation Classifier provided by scikit, which works well with the fact that each
 condition is its own discrete classification. That classifier scores all of t
he features, and then the feature selector just chooses the top 3 scores.")

X = df.iloc[:,0:279]
y = df[279]

selector = SelectKBest(mutual_info_classif, k=3)
selector.fit_transform(X,y)

print("The three most features according to the feature selector used are:",se
lector.get_support(indices=True))
print("\n")
```

Part a:
Each row corresponds to one patient, so with, 452 rows there are 452 patient
s.
Each column corresponds to one feature, so there are 280 columns and feature
s.

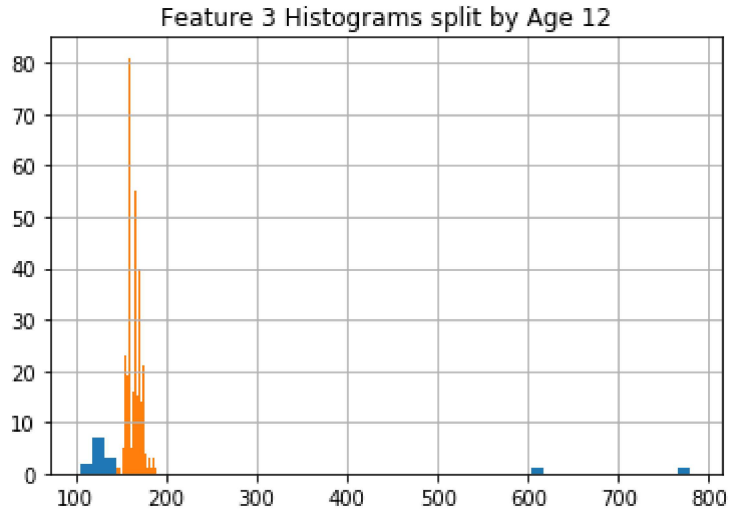
Part b:



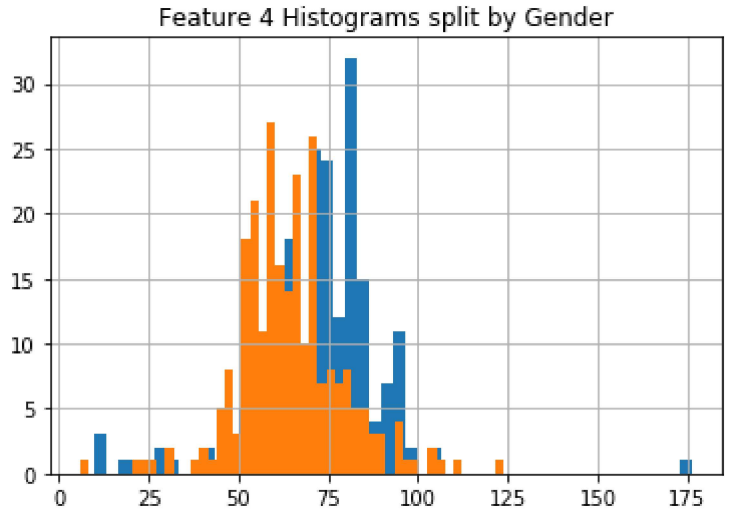
Feature 1 spans from 0-83 and is fairly normally distributed, so it most like
ly is Age



Feature 2 is a binary variable with fairly equal probability for either choic
e, so it most likely is Gender



Average Trygliceride levels Ages <12:(excluding two outliers) 124.583333333
Average Trygliceride levels Ages >=12: 164.917808219
The two points at 608 and 780 for Ages <12 can be explained by the patients n
ot fasting when the measurement was taken because triglyceride levels can spi
ke to 5-10 times the normal level when fasting
Feature 3 averages around 124 for children and then 164 for teenagers and adu
lts. This lines up with trygliceride levels where <150 mg/dL is expected for
children and adults with borderline tryglicerides have around 150-200 mg/dL.
Therefore, feature 3 is likely trygliceride levels in mg/dL



Male Average Weight(kg): 72.724137931
Female Average Weight(kg): 64.4578313253
Feature 4 when split into two data sets based on feature 1 shows that there's two distinct means, corresponding to the different genders. Those means also happen to be reasonably close to average weight in males and females, and the distributions are relatively normal. All things considered, feature 4 is most likely weight(kg).

Part d:
The way to test which features are strongly influence the patient condition i s using multi-class classification techniques to score the features. Specific ally to find the 3 most important features, we're using the Mutual Informatio n Classifier provided by scikit, which works well with the fact that each con dition is its own discrete classification. That classifier scores all of the features, and then the feature selector just chooses the top 3 scores.
The three most features according to the feature selector used are: [14 176 196]

Lab 1

1)

	$X=0$	$X=1$
$Y=0$	$1/4$	$1/4$
$Y=1$	$1/6$	$1/3$

a) $P(X=1) = \frac{1}{4} + \frac{1}{3} = \boxed{\frac{7}{12}}$

b) $P(X|Y) = \frac{P(X \cap Y)}{P(Y)} \Rightarrow \frac{P(X=1 \cap Y=1)}{P(Y=1)} = \frac{\frac{1}{3}}{\frac{1}{6} + \frac{1}{3}} = \boxed{\frac{2}{3}}$

c) $E[X] = \sum_x x f(x=x) = 1(P(X=1)) + 0(P(X=0)) = \frac{7}{12}$

$Var(X) = E[|X - E[X]|^2] = \sum_x (x - \frac{7}{12})^2 P(X=x)$

$= (1-\frac{7}{12})^2(\frac{7}{12}) + (0-\frac{7}{12})^2(\frac{5}{12})$

$= \frac{175}{1728} + \frac{245}{1728}$

$= \boxed{\frac{35}{144}}$

d) $E[X|Y=1] = \sum_x x P(X=x|Y=1) = 1(\frac{2}{3}) + 0(\frac{1}{3}) = \frac{2}{3}$

$Var(X|Y=1) = E[(X - E[X|Y=1])^2 | Y=1] = \sum_x (x-\frac{2}{3})^2 P(X=x|Y=1)$

$= (1-\frac{2}{3})^2(\frac{2}{3}) + (0-\frac{2}{3})^2(\frac{1}{3})$

$= \boxed{\frac{2}{9}}$

e) $E[X^3 + X^2 + 3Y^7 | Y=1] = E[X^3|Y=1] + E[X^2|Y=1] + 3E[Y^7|Y=1]$

$= \sum_x x^3 P(X=x|Y=1) + \sum_x x^2 P(X=x|Y=1) + 3E[Y^7|Y=y]$

$= 1^3(\frac{1}{3}) + 1^3(\frac{1}{6}) + 1^2(\frac{1}{3}) + 1^2(\frac{1}{6}) + 3$

$= \boxed{4}$

2) $v_1 = [1,1,1]$ $\quad P_1 = [3,3,3]$

$v_2 = [1,0,0]$ $\quad P_2 = [1,2,3]$

$P_3 = [0,0,1]$

Let $l_2 = v_2 = [1,0,0]$

Let $l_1 = v_1 - \text{Proj}_{l_2} v_1 = v_1 - \frac{\langle l_2, v_1 \rangle}{\langle v_1, v_1 \rangle} v_1$

$= [1,1,1] - [1,0,0]$

$= [0,1,1]$

$V = \text{span}\{l_1, l_2\}$

$\text{Proj}_{P_1} V = \frac{\langle P_1, l_1 \rangle}{\langle l_1, l_1 \rangle} l_1 + \frac{\langle P_1, l_2 \rangle}{\langle l_2, l_2 \rangle} l_2 = \frac{6}{2} l_1 + 3 l_2 = [0,3,3] + [3,0,0] = [3,3,3]$

$\text{Proj}_{P_2} V = \frac{\langle P_2, l_1 \rangle}{\langle l_1, l_1 \rangle} l_1 + \frac{\langle P_2, l_2 \rangle}{\langle l_2, l_2 \rangle} l_2 = \frac{5}{2} l_1 + l_2 = [0, \frac{5}{2}, \frac{5}{2}] + [1,0,0] = [1, \frac{5}{2}, \frac{5}{2}]$

$\text{Proj}_{P_2} V = \frac{\langle P_3, l_1 \rangle}{\langle l_1, l_1 \rangle} l_1 + \frac{\langle P_3, l_2 \rangle}{\langle l_2, l_2 \rangle} l_2 = \frac{1}{2} l_1 + 0 l_2 = [0, \frac{1}{2}, \frac{1}{2}]$

3. Central Limit Theorem:

$$\frac{S_n - n\mu}{\sigma\sqrt{n}}$$

$$P(\text{Head}) = \frac{2}{3}$$

Let X_i be the random variable that a head is flipped.

$X = 1$ if heads, 0 otherwise.

Let $S_n = X_1 + X_2 + \ldots X_n$

$$\text{Variance} = \left(\frac{2}{3}\right)\left(\frac{1}{3}\right) = \frac{2}{9}$$

Using CLT: $$\frac{50 - 100\left(\frac{2}{3}\right)}{\sqrt{100}\sqrt{2/9}}$$

$$P\left(Z_{100} < -3.54\right) = \Phi(-3.54)$$

Using Python and scipy:

```
import scipy.stats
scipy.stats.norm.cdf(-3.54)
```

Output: 0.0002 = 0.02%