```python
import numpy as np
from statistics import mean, variance, stdev
from math import sqrt
output_text = 'Homework 8 Output\n\n'

# Extract data from csv file, numbers are hours until depletion of batteries
s1 = []
s2 = []
s3 = []
s4 = []
s5 = []
n = 23
with open('HW8_DATA.csv', 'r') as f:
    lines = f.readlines()[1:]
    for line in lines:
        ele = line.split(',')
        s1.append(float(ele[0]))
        s2.append(float(ele[1]))
        s3.append(float(ele[2]))
        s4.append(float(ele[3]))
        s5.append(float(ele[4]))

# Calculate means
m1 = mean(s1)
m2 = mean(s2)
m3 = mean(s3)
m4 = mean(s4)
m5 = mean(s5)

# Calculate standard deviation
std1 = stdev(s1)
std2 = stdev(s2)
std3 = stdev(s3)
std4 = stdev(s4)
std5 = stdev(s5)

# Calculate intervals for question 1
z = 1.714
CI1_1 = [m1 + z*(std1/sqrt(n)), m1 - z*(std1/sqrt(n))]
CI1_2 = [m2 + z*(std2/sqrt(n)), m2 - z*(std2/sqrt(n))]
CI1_3 = [m3 + z*(std3/sqrt(n)), m3 - z*(std3/sqrt(n))]
CI1_4 = [m4 + z*(std4/sqrt(n)), m4 - z*(std4/sqrt(n))]
CI1_5 = [m5 + z*(std5/sqrt(n)), m5 - z*(std5/sqrt(n))]
CI1 = [CI1_1, CI1_2, CI1_3, CI1_4, CI1_5]
output_text += 'Problem 1\n'
for i in CI1:
    output_text += f'Sample {1+CI1.index(i)}: {round(i[1], 4)} to {round(i[0], 4)} feet\n'
output_text += '\n'

# Find new list of boards given defect lengths
min = 24.97
max = 25.04
samples = [s1, s2, s3, s4, s5]
s_1 , s_2 , s_3 , s_4 , s_5 = [], [], [], [], []
for length in s1:
    if length < min or length > max:
        s_1.append(length)
for length in s2:
    if length < min or length > max:
        s_2.append(length)
for length in s3:
    if length < min or length > max:
        s_3.append(length)
for length in s4:
    if length < min or length > max:
        s_4.append(length)
for length in s5:
    if length < min or length > max:
        s_5.append(length)
n_1 = len(s_1)
n_2 = len(s_2)
n_3 = len(s_3)
n_4 = len(s_4)
n_5 = len(s_5)

# Calculate intervals for question 2
z1, z2, z3, z4, z5 = 1.714, 1.714, 1.714, 1.714, 1.714
p1 = n_1/n
p2 = n_2/n
p3 = n_3/n
```

```python
p4 = n_4/n
p5 = n_5/n
CI2_1 = [p1 + z1*sqrt(p1*(1-p1)/n), p1 - z1*sqrt(p1*(1-p1)/n)]
CI2_2 = [p2 + z2*sqrt(p2*(1-p2)/n), p2 - z2*sqrt(p2*(1-p2)/n)]
CI2_3 = [p3 + z3*sqrt(p3*(1-p3)/n), p3 - z3*sqrt(p3*(1-p3)/n)]
CI2_4 = [p4 + z4*sqrt(p4*(1-p4)/n), p4 - z4*sqrt(p4*(1-p4)/n)]
CI2_5 = [p5 + z5*sqrt(p5*(1-p5)/n), p5 - z5*sqrt(p5*(1-p5)/n)]
CI2 = [CI2_1, CI2_2, CI2_3, CI2_4, CI2_5]
output_text += 'Problem 2\n'
for i in CI2:
    output_text += f'Sample {1+CI2.index(i)}: {round(i[1], 4)} to {round(i[0], 4)}\n'
output_text += '\n'

# Calculate intervals for question 2
sv1 = variance(s1)
sv2 = variance(s2)
sv3 = variance(s3)
sv4 = variance(s4)
sv5 = variance(s5)
chi_up = 33.92      # 0.050
chi_low = 12.34     # 0.950
CI3_1 = [(n-1)*sv1/chi_low, (n-1)*sv1/chi_up]
CI3_2 = [(n-1)*sv2/chi_low, (n-1)*sv2/chi_up]
CI3_3 = [(n-1)*sv3/chi_low, (n-1)*sv3/chi_up]
CI3_4 = [(n-1)*sv4/chi_low, (n-1)*sv4/chi_up]
CI3_5 = [(n-1)*sv5/chi_low, (n-1)*sv5/chi_up]
CI3 = [CI3_1, CI3_2, CI3_3, CI3_4, CI3_5]
output_text += 'Problem 3\n'
for i in CI2:
    output_text += f'Sample {1+CI2.index(i)}: {round(i[1], 4)} to {round(i[0], 4)} ft^2\n'
output_text += '\n'

# Question 4
m = 24.985      # mean
std = 0.021     # standard deviation

# Part b
flagb = []
for sample in samples:
    minimum = np.min(sample)
    maximum = np.max(sample)
    if minimum <= m <= maximum:
        flagb.append(1)

# Part c
flagc = [1, 1, 1, 1]

# Part d
flagd = []
for sample in samples:
    var_calc = sum([(i-m)**2 for i in sample])/(n-1)
    CI = CI3[samples.index(sample)]
    if CI[1] <= var_calc <= CI[0]:
        flagd.append(1)

output_text += 'Problem 4\n'
output_text += f'B) {len(flagb)}/5\n'
output_text += f'C) {len(flagc)}/5\n'
output_text += f'D) {len(flagd)}/5\n'

print(output_text)

with open('HW8_ouput.txt', 'w+') as f:
    f.write(output_text)
    f.close()
```