Sebastian's Weight and Proofs

Find Weight

```
import random
from math import sqrt

def mean(data):
    return sum(data)/len(data)

def variance(data):
    mu=mean(data)
    return sum([(x-mu)**2 for x in data])/len(data)

def stddev(data):
    return sqrt(variance(data))



weight=[80.,85,200,85,69,65,68,66,85,72,85,82,65,105,75,80,
    70,74,72,70,80,60,80,75,80,78,63,88.65,90,89,91,1.00E+22,
    75,75,90,80,75,-1.00E+22,-1.00E+22,-1.00E+22,86.54,67,70,92,70,76,81,93,
    70,85,75,76,79,89,80,73.6,80,80,120,80,70,110,65,80,
    250,80,85,81,80,85,80,90,85,85,82,83,80,160,75,75,
    80,85,90,80,89,70,90,100,70,80,77,95,120,250,60]

print mean(weight)



def calculate_weight(data, z):
    # remove outliers
    # extract data between lower and upper quartile
    # fit Gaussian using MLE
    # compute x that corresponds to standard score z

    # sort data
    sorted_data = sorted(data)
    # remove outliers where lower and upper quartiles are 23 and 70
    interquartile = sorted_data[23:70]
    # standard score z = (x-mu)/std
    # find x from the above equation: x = z*std + mu
    x = z*stddev(interquartile)+mean(interquartile)
    return x
```

Output:

```
-2.10526315789e+20 72.8098804219
```