Problem To Solve

Fred is a very predictable man. For instance, when he uses his laptop, all he does is watch TV shows. He keeps on watching TV shows until his battery dies. Also, he is a very meticulous man, i.e. he pays great attention to minute details. He has been keeping logs of every time he charged his laptop, which includes how long he charged his laptop for and after that how long was he able to watch the TV. Now, Fred wants to use this log to predict how long will he be able to watch TV for when he starts so that he can plan his activities after watching his TV shows accordingly.

Challenge

You are given access to Fred's laptop charging log by reading from the file "trainingdata.txt". The training data file will consist of 100 lines, each with 2 comma-separated numbers.

The first number denotes the amount of time the laptop was charged. The second number denotes the amount of time the battery lasted. The training data file can be downloaded here (this will be the same training data used when your program is run). The input for each of the test cases will consist of exactly 1 number rounded to 2 decimal places. For each input, output 1 number: the amount of time you predict his battery will last.

Sample Input

1.50

Sample Output

3.00

Scoring

Your score will be 10-X, where X is the sum of the distances you are from expected answer of each test case. For instance if there are 2 test cases with expected answer 4 and you print 3 for the first one and 6 for the second one your score will be 10-(1+2) = 7.

Solution

First import the necessary libraries.
pandas : for data manipulation and analysis.
numpy : for working in domain of linear algebra, fourier transform, and matrices.

In [ ]:

```
import math
import os
import random
import re
import sys
import pandas
import numpy
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
```

reading data from the text file

In [ ]:
```
data = pandas.read_csv('trainingdata.txt',sep=',')
```

extraction the max value of the hours can the computer run without charging then make a new data set just for the overfitting values to extract them from the original data set when we need to train our model

In [ ]:
```
max_value = data.iloc[:, 1].max()
data2 = data[data.iloc[:, 1] == max_value]
data3 = data[data.iloc[:, 1] < max_value]

min_value = data2.iloc[:, 0].min()
```

reshaped in such a way that the resulting array of data has only 1 column

In [ ]:
```
x= data3.iloc[:, 0].values.reshape(-1, 1)
y= data3.iloc[:, 1].values.reshape(-1, 1)
```

start training our model then generate a linear result to predict future values

In [ ]:
```
xtrain, xtest, ytrain, ytest = train_test_split(x,y,random_state=1)
model = LinearRegression().fit(xtrain,ytrain)
```

predict a value insered by the user

In [ ]:
```
cb = float(input())
if cb >= min_value :
   print(max_value)
else :
    y_pred = model.predict([[cb]])
    c = float(y_pred)
    print('',c)
```