

## . \_

# ☆ Predicting the Temperature



Given the hourly temperature data for each 24 hour period in p prior days spanning from startDate to endDate inclusive, predict the hourly temperature data for the next n days starting the first hour after endDate.

## 2

### **Function Description**

Complete the function *predictTemperature* in the editor below. The function must return an array of floating-point numbers, one predicted temperature for each hour of *n* days immediately following *endDate* in chronological order.



predictTemperature has the following parameter(s):

startDate: a string in the format yyyy-mm-dd

endDate: a string in the format yyyy-mm-dd

temperature[temperature[0],...temperature[(24\*p)-1]]: an array floating-point numbers temperature[i] which represent the temperature at each yyyy-mm-dd hh:00 timestamp in the inclusive range from startDate to endDate.

n: the integer number of days to predict

#### **Constraints**

- 2013-01-01 ≤ startDate ≤ endDate ≤ 2015-01-01
- $1 \le p \le 154$
- 1 ≤ n ≤ 48

#### **Evaluation**

- The predicted temperature at a timestamp is considered to be correct if the absolute difference between the actual and predicted temperatures is not greater than 5° C.
- The accuracy of the prediction is defined as:  $\frac{\text{(Total number of correct predictions)}}{(24 \times n)^2}$
- The score for each test case is calculated as: accuracy × (test case weight).
- The final score is the sum of all test case scores.

**Input Format for Custom Testing** 

Sample Case 0

#### Sample Input 0

2013-01-01

2013-01-01

24

34.38

34.36

34.74

35.26 35.23

35.29

35.64

36.02

36.1 36.98

37.01

36.75

36.01 35.66

34.72

33.9

32.62

31.51 30.73

29.5 26.94

25.47

23.84

22.55

### 1

### **Sample Output**

36.02

36.1

36.98

37.01

36.75 36.01

35.66

34.72

33.9 32.62



25.47 23.84 22.55 21.03 19.92 18.77 18.48 18.07 17.91 17.11

## **Explanation**

Given the hourly temperature data for the  $24 \times p = 24 \times 1 = 24$  hour period starting on **2013–01–01**, the task is to predict the hourly temperature data for the  $24 \times n = 24 \times 1 = 24$  hour period starting on 2013-01-02. The table below depicts sample predictions for the given temperature data that contains a total of 19 correct predictions. The accuracy of these predictions is  $^{19}/_{24} = 0.79$  and, because the test case weight is 5, the total score for this test case is  $0.79 \times 5 = 3.95$ .

**O** 20m

to test end

TEMPERATURE				
Timestamp	Temperature	Predicted Temperature	Temperature Difference	Correct Prediction?
2013-01-01 00:00	36.02	35.17	0.85	Correct
2013-01-01 01:00	36.1	34.47	1.63	Correct
2013-01-01 02:00	36.98	38.38	-1.40	Correct
2013-01-01 03:00	37.01	41.22	-4.21	Correct
2013-01-01 04:00	36.75	40.09	-3.34	Correct
2013-01-01 05:00	36.01	33.57	2.44	Correct
2013-01-01 06:00	35.66	36.83	-1.17	Correct
2013-01-01 07:00	34.72	35.04	-0.32	Correct
2013-01-01 08:00	33.9	31.96	1.94	Correct
2013-01-01 09:00	32.62	33.41	-0.79	Correct
2013-01-01 10:00	31.51	27.69	3.82	Correct
2013-01-01 11:00	30.73	28.74	1.99	Correct
2013-01-01 12:00	29.5	29.91	-0.41	Correct
2013-01-01 13:00	26.94	30.68	-3.74	Correct
2013-01-01 14:00	25.47	30.46	-4.99	Correct
2013-01-01 15:00	23.84	22.52	1.32	Correct
2013-01-01 16:00	22.55	17.48	5.07	Incorrect
2013-01-01 17:00	21.03	27.24	-6.21	Incorrect
2013-01-01 18:00	19.92	25.12	-5.20	Incorrect
2013-01-01 19:00	18.77	23.89	-5.12	Incorrect
2013-01-01 20:00	18.48	21.46	-2.98	Correct
2013-01-01 21:00	18.07	22.54	-4.47	Correct
2013-01-01 22:00	17.91	23.15	-5.24	Incorrect
2013-01-01 23:00	17.11	18.85	-1.74	Correct

For help on how to read input and write output in Python 3, click here.

**(3)** 

#!/bin/python3 ··· 1

10

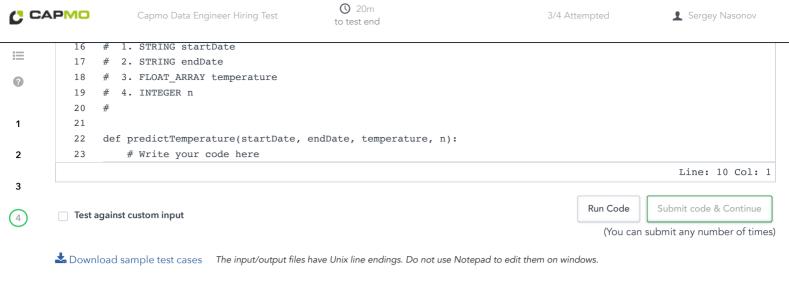
11

Draft saved 04:54 pm

# Complete the 'predictTemperature' function below. 12

Python 3

Original Code



About Privacy Policy Terms of Service