



You can view this report online at : <https://www.hackerrank.com/x/tests/1667237/candidates/55118491/report>

Full Name:

Ayesha Saeed

Email:

as07519@st.habib.edu.pk

Test Name:

CS 316 Lab # 1 Fall 2023

Taken On:

23 Aug 2023 15:31:17 PKT

Time Taken:

99 min 45 sec/ 120 min

Work Experience:

2 years

Invited by:

Muhammad Munawwar

Invited on:

23 Aug 2023 15:25:23 PKT

Skills Score:

Tags Score:

Python 3

450/450

np

0/150

numpy

450/600

81.8%

1350/1650

scored in CS 316 Lab # 1 Fall 2023 in 99 min 45 sec on 23 Aug 2023 15:31:17 PKT

Recruiter/Team Comments:

No Comments.

	Question Description	Time Taken	Score	Status
Q1	Fresh Start: Zero Tensor > Multiple Choice	3 min 14 sec	0/ 150	✗
Q2	Fraud at Chocomo factory > Multiple Choice	4 min 25 sec	150/ 150	✓
Q3	Chocomo Factory > Multiple Choice	7 min 8 sec	0/ 150	✗
Q4	Generating Matrices > Multiple Choice	4 min 6 sec	150/ 150	✓
Q5	Summing up N natural numbers. > Coding	6 min 42 sec	150/ 150	✓
Q6	Fahrenheit to Celsius > Coding	2 min 16 sec	150/ 150	✓
Q7	YauTube: Broadcast your agenda > Coding	40 min 3 sec	150/ 150	✓
Q8	Matrix Multiplication > Coding	5 min 56 sec	150/ 150	✓
Q9	Exponents are Evil! > Coding	16 min 17 sec	150/ 150	✓
Q10	Grocery Bill > Coding	5 min 31 sec	150/ 150	✓
Q11	Generating first N natural numbers. > Coding	2 min 35 sec	150/ 150	✓

Wrong Answer

Score 0

Fresh Start: Zero Tensor > Multiple Choice

QUESTION DESCRIPTION

Which of these commands will you run in order to create a 28 x 28 x 30 zero tensor in numpy?

CANDIDATE ANSWER

Options: (Expected answer indicated with a tick)

☐ np.zero(28, 28, 30)

☒ np.zero((28, 28, 30))

☐ np.zeros(28, 28, 30)

☐ np.zeros(30, 28, 28)

☒ np.zeros((28, 28, 30))

No Comments

Correct Answer

Score 150

Fraud at Chocomo factory > Multiple Choice

QUESTION DESCRIPTION

You look at the data you generated and you saw that what you got was vastly less. You inform this to the Chocomo foundation which conducts an audit and finds out that one of the employees, Augustus Gloop, has been eating the chocolate rather than putting it in the biscuits.

Each employee is supposed to mention how much chocolate they used in each shift. The data is stored in a 2d numpy array `chocolate_used`. where each row refers to each employee and a column refers to each shift.

You decide to fire Augustus Gloop and set his `chocolate_used` entries for every shift as 0. Augustus's entry is stored in the last row.

How would you do this in numpy? (Choose all that apply)

CANDIDATE ANSWER

Options: (Expected answer indicated with a tick)

☐ `chocolate_used[: , -1] = 0`

☒ `chocolate_used[-1] = 0`

☒ `chocolate_used[-1 , :] = 0`

☐ `chocolateused[6 , :-1]`

No Comments

Wrong Answer

Score 0

Chocomo Factory > Multiple Choice

QUESTION DESCRIPTION

Chocomo is a chocolate-filled biscuit company. You have a feeling they don't use enough chocolate. So, like any *normal* human, you phone the production manager and ask how much chocolate they put in, and the

2/15

manager, like any *normal* human, tells you that the amount of chocolate in our 100 biscuits pack follows a **normal distribution with a mean of 50g and a standard deviation of 9g.**

You decide to simulate the Chocomo plant to determine if you are just unlikely or not. How would you generate *100 samples* of biscuits using numpy, with each element in the array indicating how much chocolate each pack contains? (Please choose all that apply.)

CANDIDATE ANSWER

Options: (Expected answer indicated with a tick)

- ☐ np.random.normal(50,3,100)
- ☐ np.random.normal(100,50,3)
- ☒ np.random.normal(50,9,size=100)
- ☒ np.random.normal(50,9,100)

No Comments

QUESTION 4



Correct Answer

Score 150

Generating Matrices > Multiple Choice

QUESTION DESCRIPTION

How can you generate the following matrix using numpy? (Choose all that apply)

$$\begin{bmatrix} 1 & 2 & 2 \\ 2 & 1 & 2 \\ 2 & 2 & 1 \end{bmatrix}$$

CANDIDATE ANSWER

Options: (Expected answer indicated with a tick)

- ☐ np.one(3 , 3) * 2 - np.eye(3)
- ☐ np.ones((3 , 3))
- ☒ 2*np.ones((3 , 3))- np.eye(3)
- ☒ np.subtract (np.multiply (2 , np.ones((3 , 3))) , np.eye(3))

No Comments

QUESTION 5



Correct Answer

Score 150

Summing up N natural numbers. > Coding

Python 3

numpy

QUESTION DESCRIPTION

In the previous question, we learned how to use **Numpy** to generate **n** natural numbers. In this question, we will build on that to find the **sum** of those n natural numbers.

Given n = 50, the sum will be 1275.

Function Description

In the editor below, complete the function sumOfNNaturalNumbers. The function must return an integer sum as its result.

sumOfNNaturalNumbers has the following parameter(s):

n: an integer

Constraints

• $n \geq 1$

▼ Input Format For Custom Testing

The first line contains an integer, *n*, denoting the number of natural numbers we want to sum.

▼ Sample Case 0

Sample Input For Custom Testing

```
5
```

Sample Output

```
15
```

Explanation

$1 + 2 + 3 + 4 + 5 = 15$

Sample Case 1

Sample Input For Custom Testing

```
50
```

Sample Output

```
1275
```

Explanation

$1 + 2 + 3 + \dots + 50 = 1275$

CANDIDATE ANSWER

Language used: Python 3

```
1
2 #
3 # Complete the 'sumOfNNaturalNumbers' function below.
4 #
5 # The function is expected to return an INTEGER.
6 # The function accepts INTEGER n as parameter.
7 #
8
9 def sumOfNNaturalNumbers(n):
10     X = np.arange(n+1)
11     s = X.sum(keepdims=True)
12     return s[0]
13
```

TESTCASE	DIFFICULTY	TYPE	STATUS	SCORE	TIME TAKEN	MEMORY USED
TestCase 0	Easy	Sample case	✔ Success	10	0.4351 sec	30.9 KB
TestCase 1	Easy	Sample case	✔ Success	10	0.3277 sec	30.7 KB
TestCase 2	Easy	Hidden case	✔ Success	10	0.3513 sec	30.9 KB
TestCase 3	Easy	Hidden case	✔ Success	10	0.2152 sec	30.9 KB

TestCase 4	Easy	Hidden case	✓	Success	10	0.2119 sec	30.6 KB
TestCase 5	Easy	Hidden case	✓	Success	10	0.3129 sec	30.8 KB
TestCase 6	Easy	Hidden case	✓	Success	10	0.7471 sec	31 KB
TestCase 7	Easy	Hidden case	✓	Success	10	0.5146 sec	31 KB
TestCase 8	Easy	Hidden case	✓	Success	10	0.3845 sec	30.9 KB
TestCase 9	Easy	Hidden case	✓	Success	10	0.2275 sec	30.8 KB
TestCase 10	Easy	Hidden case	✓	Success	10	0.414 sec	30.6 KB
TestCase 11	Easy	Hidden case	✓	Success	10	0.4251 sec	30.6 KB
TestCase 12	Easy	Hidden case	✓	Success	10	0.2738 sec	30.6 KB
TestCase 13	Easy	Hidden case	✓	Success	10	0.4056 sec	30.9 KB
TestCase 14	Easy	Hidden case	✓	Success	10	0.2474 sec	30.9 KB

No Comments

QUESTION 6



Correct Answer

Score 150

Fahrenheit to Celsius > Coding

QUESTION DESCRIPTION

To make a blog post about the unusual weather in Texas, USA, you use an API to obtain weather data. Unfortunately, the data is in Fahrenheit, which you are not familiar to, therefore you need to convert it to Celsius in order to do some analysis in your essay.

Convert a numpy array of Fahrenheit temperature data to Celsius to two decimal places. (You may use `np.round(array,decimal)`).

For example, if you have the numpy array `[32,37,43,46]`, the output will be `[0., 2.78, 6.11, 7.78]`
Since

$$C = (F - 32) \times \frac{5}{9}$$

Function Description

Complete the function `fahrenheitToCelsius` in the editor below. The function must state what must be returned or printed.

`fahrenheitToCelsius` has the following parameter(s):

`fahrenheit`: a 1D numpy array of floats.

▼ Input Format For Custom Testing

The input contains space separated values where each value is a particular reading.

▼ Sample Case 0

Sample Input For Custom Testing

```
32 37 43 46
```

Sample Output

```
0.0 2.78 6.11 7.78
```

Explanation

We use the formula mentioned above and round it to 2 decimal places.

CANDIDATE ANSWER

Language used: **Python 3**

```

1
2 #
3 # Complete the 'celsiusToFahrenheit' function below.
4 #
5 # The function is expected to return a numpy array.
6 # The function accepts numpy array as parameter.
7 #
8
9 def fahrenheitToCelsius(fahrenheit):
10     cel = (fahrenheit - 32) * (5/9)
11     cel = np.round(cel,2)
12     return cel
13
14

```

TESTCASE	DIFFICULTY	TYPE	STATUS	SCORE	TIME TAKEN	MEMORY USED
TestCase 0	Easy	Sample case	✓ Success	10	0.3971 sec	30.8 KB
TestCase 1	Easy	Hidden case	✓ Success	10	0.6303 sec	30.8 KB
TestCase 2	Easy	Hidden case	✓ Success	10	0.3212 sec	30.9 KB
TestCase 3	Easy	Hidden case	✓ Success	10	0.2821 sec	30.5 KB
TestCase 4	Easy	Hidden case	✓ Success	10	0.2605 sec	30.9 KB
TestCase 5	Easy	Hidden case	✓ Success	10	0.3424 sec	30.3 KB
TestCase 6	Easy	Hidden case	✓ Success	10	0.2131 sec	30.8 KB
TestCase 7	Easy	Hidden case	✓ Success	10	0.3412 sec	30.6 KB
TestCase 8	Easy	Hidden case	✓ Success	10	0.3578 sec	30.6 KB
TestCase 9	Easy	Hidden case	✓ Success	10	0.2441 sec	30.9 KB
TestCase 10	Easy	Hidden case	✓ Success	10	0.3467 sec	30.9 KB
TestCase 11	Easy	Hidden case	✓ Success	10	0.2562 sec	30.8 KB
TestCase 12	Easy	Hidden case	✓ Success	10	0.3622 sec	30.6 KB
TestCase 13	Easy	Hidden case	✓ Success	10	0.6382 sec	30.5 KB
TestCase 14	Easy	Hidden case	✓ Success	10	0.28 sec	30.3 KB

No Comments

QUESTION 7



Correct Answer

Score 150

YouTube: Broadcast your agenda > Coding

QUESTION DESCRIPTION

The contents of this question are fictional and satirical.

You work at **Youtube**, a video-sharing network. Since times are bad, you decide to sell out to politicians and billionaires and promote videos that promote their agenda. Several prominent politicians opt to hire you. Your recommendation algorithm provides a score for each video for each user. To promote a video,

simply boost the score for that video for each user. You also provide the service of changing the landscape of the entire platform for billionaires by changing the score for each video.

E.g. If you have 2 users and 3 videos, the landscape of your platform is

$$\begin{bmatrix} 3 & 4 & 1 \\ -1 & 2 & 3 \end{bmatrix}$$

where each row represents a user and each column represents a video.

If a **politician** asks you to promote the first video by 10 points, the resulting landscape will be

$$\begin{bmatrix} 13 & 4 & 1 \\ 9 & 2 & 3 \end{bmatrix}$$

If a **billionaire** wants to reduce the score of video 1 by 10, keep it constant for 2, and increase video 3 by 20. His order is

$$\begin{bmatrix} -10 & 0 & 20 \end{bmatrix}$$

And hence the resulting landscape will be

$$\begin{bmatrix} -7 & 4 & 21 \\ -11 & 2 & 23 \end{bmatrix}$$

Your goal is to implement 2 functions **using broadcasting** that allow for these changes.

Function Descriptions

Complete the function *billionaire_influence* in the editor below. The function must return the resulting landscape of your platform.

billionaire_influence has the following parameter(s):

***current_landscape*: A 2d numpy array of the current landscape**

***demands*: A 1d array, signifying how much to change the score of each video by.**

Complete the function *politician_influence* in the editor below. The function must return the resulting landscape of your platform.

politician_influence has the following parameter(s):

***current_landscape*: A 2d numpy array of the current landscape**

***index*: Index of the video they want to promote**

***update_amount*: Amount to update the score by**

▼ Input Format For Custom Testing

The first line contains 3 numbers, space-separated. The first number specifies whether you want to run *politician_influence* (0) or *billionaire_influence* (1). The second number represents the number of users (m) and the third number represents the number of videos (n).

The next m lines will contain n space-separated values representing each user.

The last line will either contain 2 values: index and update_amount (space separated) if *politician_influence* is run otherwise n values specifying the billionaire's demand

▼ Sample Case 0

Sample Input For Custom Testing

```
0 2 3
3 4 1
-1 2 3
0 10
```

Sample Output

```
13 4 1
9 2 3
```

▼ Sample Case 1

Sample Input For Custom Testing

```
1 2 3
3 4 1
-1 2 3
-10 0 20
```

Sample Output

```
-7 4 21
-11 2 23
```

CANDIDATE ANSWER

Language used: **Python 3**

```
1 def billionaire_influence(current_landscape, demands):
2     res = current_landscape + demands
3     return res
4
5
6
7
8
9 def politician_influence(current_landscape, index, update_amount):
10     # Hint : Column Broadcast
11     # write your code here
12
13     current_landscape[:,index] += update_amount
14     return current_landscape
15
16
17
```

TESTCASE	DIFFICULTY	TYPE	STATUS	SCORE	TIME TAKEN	MEMORY USED
TestCase 0	Easy	Sample case	✔ Success	20	0.5417 sec	30.8 KB
TestCase 1	Easy	Sample case	✔ Success	20	0.3671 sec	30.7 KB
TestCase 2	Easy	Hidden case	✔ Success	10	0.2093 sec	30.6 KB
TestCase 3	Easy	Hidden case	✔ Success	20	0.2128 sec	30.7 KB
TestCase 4	Easy	Hidden case	✔ Success	20	0.4424 sec	30.5 KB
TestCase 5	Easy	Hidden case	✔ Success	20	0.2653 sec	30.7 KB
Testcase 6	Easy	Sample case	✔ Success	20	0.3302 sec	30.7 KB
Testcase 7	Easy	Sample case	✔ Success	20	0.2148 sec	30.9 KB

No Comments



Correct Answer

Score 150

QUESTION DESCRIPTION

Matrix **A** with dimensions $m \times n$ can multiply matrix **B** with dimensions $p \times q$ if and only if $n = p$ and the resulting matrix is of dimension $m \times q$.

Write a function that accepts two numpy matrices (2d arrays) and determines whether matrix multiplication is possible. If it is, return a **tuple (m,q)**. If not, return **"NOT POSSIBLE"**.

If we have a matrix **A** with dimensions 3 x 2 and a matrix **B** with dimensions 2 x 5, the function will return **(3,5)**.

However, if we have a matrix **A** with dimensions 1 x 2 and a matrix **B** with dimensions 1 x 3, the function will return **"NOT POSSIBLE"**.

Function Description

Complete the function *isMatrixMultiplicationPossible* in the editor below. If matrix multiplication is possible, it returns a **tuple (m,q)** indicating the dimension of the resultant matrix when A and B are multiplied; otherwise, it returns the **string "NOT POSSIBLE"**.

isMatrixMultiplicationPossible has the following parameter(s):

A: a 2D numpy array

B: a 2D numpy array

▼ Input Format For Custom Testing

The first line contains two space separated values that represent the number of rows and columns in Matrix A.

The second line contains space separated values for each of the matrix's values (row by row).

The third line contains two space separated values that represent the number of rows and columns in Matrix B.

The fourth line contains space separated values for each of the matrix's values (row by row).

▼ Sample Case 0

Sample Input For Custom Testing

```
1 2
0 0
1 3
5 6 7
```

Sample Output

```
NOT POSSIBLE
```

Explanation

We have two matrices

$$A = \begin{bmatrix} 0 & 0 \end{bmatrix}, B = \begin{bmatrix} 5 & 6 & 7 \end{bmatrix}$$

Matrix multiplication is not possible because A has dimensions 1 x 2 and B has dimensions 1 x 3.

▼ Sample Case 1

Sample Input For Custom Testing

```
1 2
3 4
2 4
5 7 8 6 7 7 7 7
```

Sample Output

```
(1, 4)
```

Explanation

$$A = \begin{bmatrix} 3 & 4 \end{bmatrix}, B = \begin{bmatrix} 5 & 7 & 8 & 6 \\ 7 & 7 & 7 & 7 \end{bmatrix}$$

Given that A has dimensions of 1 x 2 and B has dimensions of 2 x 4, matrix multiplication is possible, and the resulting matrix is 1 x 4.

CANDIDATE ANSWER

Language used: **Python 3**

```

1  #
2  # Complete the 'isMatrixMultiplicationPossible' function below.
3  #
4  # The function accepts 2 numpy arrays as parameter A,B respectively.
5  #
6
7  def isMatrixMultiplicationPossible(A,B):
8      # Write your code here
9      a = A.shape
10
11     b= B.shape
12
13     if a[1]==b[0]:
14         return ((a[0],b[1]))
15     return 'NOT POSSIBLE'

```

TESTCASE	DIFFICULTY	TYPE	STATUS	SCORE	TIME TAKEN	MEMORY USED
TestCase 0	Easy	Sample case	✓ Success	10	0.2232 sec	31 KB
TestCase 1	Easy	Sample case	✓ Success	10	0.3589 sec	30.4 KB
TestCase 2	Easy	Hidden case	✓ Success	10	0.5515 sec	32.5 KB
TestCase 3	Easy	Hidden case	✓ Success	10	0.3976 sec	31.3 KB
TestCase 4	Easy	Hidden case	✓ Success	10	0.5782 sec	31.8 KB
TestCase 5	Easy	Hidden case	✓ Success	10	0.3745 sec	32.6 KB
TestCase 6	Easy	Hidden case	✓ Success	10	0.2941 sec	32.6 KB
TestCase 7	Easy	Hidden case	✓ Success	10	0.3501 sec	33.3 KB
TestCase 8	Easy	Hidden case	✓ Success	10	0.2581 sec	31.7 KB
TestCase 9	Easy	Hidden case	✓ Success	10	0.3802 sec	32 KB
TestCase 10	Easy	Hidden case	✓ Success	10	0.2379 sec	30.1 KB
TestCase 11	Easy	Hidden case	✓ Success	10	0.2504 sec	34.4 KB
TestCase 12	Easy	Hidden case	✓ Success	10	0.2322 sec	32.7 KB
TestCase 13	Easy	Hidden case	✓ Success	10	0.3841 sec	31.5 KB
TestCase 14	Easy	Hidden case	✓ Success	10	0.2328 sec	31.4 KB

No Comments

QUESTION 9



Correct Answer

Exponents are Evil! > Coding

QUESTION DESCRIPTION

Later on, in this course, you will be introduced to the Softmax function which converts numbers into probabilities. The following is the softmax function:

$$S(y)_i = \frac{\exp(y_i)}{\sum_{j=1}^n \exp(y_j)}$$

However, the Softmax function can suffer from overflow. This is because (e^{32}) is greater than 2^{32} which is the capacity of an uint32 number. We can preprocess the input array to prevent overflow from exponentiation and simplify exponentiation.

The preprocessing steps involve **subtracting the maximum of each row of y from each element in y** . This will result in the biggest pre-exponentiated value being 0 and avoiding numerical instability.

This preprocessing step will not affect our result as shown below.

$$S(y)_i = \frac{\exp(y_i) \div \exp(\max(y))}{\sum_{j=1}^n \exp(y_j) \div \exp(\max(y))} = \frac{\exp(y_i)}{\sum_{j=1}^n \exp(y_j)}$$

Your objective in this lab is to preprocess the array rather than implement the softmax function. Assume you are given a two-dimensional array A which is equal to

$$\mathbf{A} = \begin{bmatrix} 1 & 2 & 5 \\ 10 & 20 & 30 \end{bmatrix}$$

After the preprocessing step, the resulting array will be equal to

$$\mathbf{A} = \begin{bmatrix} -4 & -3 & 0 \\ -20 & -10 & 0 \end{bmatrix}$$

In the first row, the maximum value is **5** so after subtracting 5 from each element in the first row you get **[-4 -3 0]**. Similarly, in the second row, the maximum value is 30 so after subtracting 30 from each element in the second row you get **[-20 -10 0]**.

Function Description

In the editor below, complete the function `softmax_preprocess`. The function must perform the transformation and return the array z .

`softmax_preprocess` has the following parameter(s):

y : a 2d numpy array

▼ Input Format For Custom Testing

The first line contains an integer, m, n defining the rows in the 2d array.
The next lines m lines each contain the n columns for the rows.

▼ Sample Case 0

Sample Input For Custom Testing

```
2 3
1.0 2.0 5.0
10.0 20.0 30.0
```

Sample Output

```
-4.0 -3.0 0.0
-20.0 -10.0 0.0
```

CANDIDATE ANSWER

Language used: **Python 3**

```
1
2 #
3 # Complete the 'softmax_preprocess' function below.
4 #
5 # The function is expected to return a 2D numpy array.
6 # The function accepts 2D numpy array y as parameter.
7 #
8
9 def softmax_preprocess(y):
10     rows = y.shape[0]
11     arr = np.max(y, axis=1, keepdims=True)
12     res = y - arr
13     return res
14
15
16
```

TESTCASE	DIFFICULTY	TYPE	STATUS	SCORE	TIME TAKEN	MEMORY USED
TestCase 0	Easy	Sample case	✔ Success	30	0.2318 sec	30.7 KB
TestCase 1	Easy	Hidden case	✔ Success	20	0.4318 sec	31 KB
TestCase 2	Easy	Hidden case	✔ Success	20	0.2675 sec	30.9 KB
TestCase 3	Easy	Hidden case	✔ Success	20	0.4519 sec	30.4 KB
TestCase 4	Easy	Hidden case	✔ Success	20	0.2605 sec	31.1 KB
TestCase 5	Easy	Hidden case	✔ Success	20	0.3543 sec	30.8 KB
Testcase 6	Easy	Sample case	✔ Success	20	0.2512 sec	30.8 KB

No Comments

QUESTION 10



Correct Answer

Score 150

Grocery Bill > Coding

QUESTION DESCRIPTION

Your mother gives you a list of items to purchase at a grocery shop. As a person with severe social anxiety who does not want to waste the cashier's time pulling notes from your wallet, you decide to enter a vacant aisle and compute the total bill with Python.

Write a function that, given two numpy arrays, one providing the number of things and another containing the price of each item, computes the total bill.

For example, if you have the arrays [1,3] and [15,20], your bill is $1 * 15 + 3 * 20 = 75$.

Function Description

Complete the function `computeGroceryBill` in the editor below. **The function must return the total bill amount rounded to 2 decimal places.**

`computeGroceryBill` has the following parameter(s):

`item_quantity`: a numpy array

`item_price`: a numpy array

▼ Input Format For Custom Testing

The first line contains space separated values which represent the quantity of each type of item.

The second line contains space separated values which indicate the price of each item.

▼ Sample Case 0

Sample Input For Custom Testing

```
1 3
15 20
```

Sample Output

```
75.0
```

Explanation

$1 * 15 + 3 * 20 = 75$

CANDIDATE ANSWER

Language used: **Python 3**

```
1 #
2 # Complete the 'computeGroceryBill' function below.
3 #
4 # The function is expected to return a floating number.
5 # The function accepts following parameters:
6 # 1. int numpy vector item_quantity
7 # 2. float numpy vector item_price
8 #
9
10 def computeGroceryBill(item_quantity, item_price):
11     bill = np.multiply(item_quantity, item_price)
12     bill = bill.sum(keepdims=True)
13     return np.round(bill[0], 2)
14
```

TESTCASE	DIFFICULTY	TYPE	STATUS	SCORE	TIME TAKEN	MEMORY USED
TestCase 0	Easy	Sample case	✔ Success	10	0.2029 sec	30.9 KB
TestCase 1	Easy	Hidden case	✔ Success	10	0.2046 sec	31 KB
TestCase 2	Easy	Hidden case	✔ Success	10	0.2915 sec	30.9 KB
TestCase 3	Easy	Hidden case	✔ Success	10	0.3515 sec	30.6 KB
TestCase 4	Easy	Hidden case	✔ Success	10	0.2123 sec	30.7 KB
TestCase 5	Easy	Hidden case	✔ Success	10	0.5495 sec	30.8 KB
TestCase 6	Easy	Hidden case	✔ Success	10	0.3678 sec	30.8 KB
TestCase 7	Easy	Hidden case	✔ Success	10	0.2011 sec	30.6 KB
TestCase 8	Easy	Hidden case	✔ Success	10	0.3641 sec	30.5 KB
TestCase 9	Easy	Hidden case	✔ Success	10	0.2242 sec	30.4 KB

TestCase 9	Easy	Hidden case	✓	Success	10	0.3343 sec	30.4 KB
TestCase 10	Easy	Hidden case	✓	Success	10	0.3078 sec	30.7 KB
TestCase 11	Easy	Hidden case	✓	Success	10	0.2715 sec	30.7 KB
TestCase 12	Easy	Hidden case	✓	Success	10	0.2626 sec	30.9 KB
TestCase 13	Easy	Hidden case	✓	Success	10	0.1956 sec	31 KB
TestCase 14	Easy	Hidden case	✓	Success	10	0.2056 sec	30.8 KB

No Comments

QUESTION 11



Correct Answer

Score 150

Generating first N natural numbers. > Coding Python 3 numpy

QUESTION DESCRIPTION

Numpy is an extremely useful tool that we will use throughout this course. In this question, we want to generate the first n natural numbers (1,2,3,...,n).

We want to do this with numpy's **arange** function.

For example when $n = 5$, we will generate a numpy array [1,2,3,4,5] and output will be 1, 2, 3, 4, 5

Function Description

In the editor below, complete the function generateNNaturalNumbers. The function's output must be a numpy array containing the first n natural numbers.

generateNNaturalNumbers has the following parameter(s):

n : an integer

Constraints

$n \geq 1$

▼ Input Format For Custom Testing

The first line contains an integer, n , which indicates how many natural numbers we want to generate.

▼ Sample Case 0

Sample Input For Custom Testing

7

Sample Output

1, 2, 3, 4, 5, 6, 7

Explanation

The first 7 natural numbers are 1, 2, 3, 4, 5, 6, 7.

PLEASE NOTE THAT YOU JUST HAVE TO WRITTEN THE NUMPY ARRAY.

CANDIDATE ANSWER

Language used: **Python 3**

```

1 #
2 # Complete the 'generateNNaturalNumbers' function below.
3 #
4 # The function is expected to return a numpy array.
5 # The function accepts an INTEGER n as parameter.

```

```

6  #
7
8  def generateNNaturalNumbers(n):
9      arr = np.arange(n+1)
10     return arr[1:]

```

TESTCASE	DIFFICULTY	TYPE	STATUS	SCORE	TIME TAKEN	MEMORY USED
TestCase 0	Easy	Sample case	✔ Success	10	0.256 sec	30.8 KB
TestCase 1	Easy	Sample case	✔ Success	10	0.194 sec	30.4 KB
TestCase 2	Easy	Hidden case	✔ Success	10	0.4065 sec	30.7 KB
TestCase 3	Easy	Hidden case	✔ Success	10	0.5023 sec	30.5 KB
TestCase 4	Easy	Hidden case	✔ Success	10	0.3735 sec	30.6 KB
TestCase 5	Easy	Hidden case	✔ Success	10	0.2006 sec	30.8 KB
TestCase 6	Easy	Hidden case	✔ Success	10	0.4976 sec	30.7 KB
TestCase 7	Easy	Hidden case	✔ Success	10	0.1999 sec	30.7 KB
TestCase 8	Easy	Hidden case	✔ Success	10	0.5119 sec	30.6 KB
TestCase 9	Easy	Hidden case	✔ Success	10	0.4952 sec	30.8 KB
TestCase 10	Easy	Hidden case	✔ Success	10	0.3049 sec	30.7 KB
TestCase 11	Easy	Hidden case	✔ Success	10	0.6655 sec	30.5 KB
TestCase 12	Easy	Hidden case	✔ Success	10	0.6995 sec	31 KB
TestCase 13	Easy	Hidden case	✔ Success	10	0.4362 sec	30.5 KB
TestCase 14	Easy	Hidden case	✔ Success	10	0.3124 sec	30.5 KB

No Comments