Name:________________________________ Roll No:________________ Section:____________________

NATIONAL UNIVERSITY OF COMPUTER AND EMERGING SCIENCES, LAHORE CAMPUS

	Course:	**Intro. to Data Science**	**Course Code:**	**DS2001**
	Program:	**BS (Data Science)**	**Semester:**	**Fall 2021**
	Duration	**30 Minutes**	**Total Marks:**	**10**
	Paper Date:	**24-Dec-21**	**Weight:**	
	Section:	**B**	**Page(s):**	**2**
	Quiz:	**3**		

Instructions: Attempt the quiz on question paper and write concise answers.

Marks:	
Total:	10

Q1. You are given the following array

```
>>> arr = np.array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9])
```

Use a numpy method to replace all the odd numbers in the array with their squares (single line code only):

```
array([ 0,  1,  2,  9,  4, 25,  6, 49,  8, 81])
```

Any of these:

1. `arr[arr%2!=0] **= 2` *the same as* `arr[arr%2!=0] = arr[arr%2!=0] ** 2`
2. `arr = np.where(arr%2!=0,arr**2,arr)`

Q2. When trying stack the following two arrays horizontally, an error was encountered:

```
>>> a = np.arange(10, 20)
>>> b = np.array([[0,1,2,3,4],[5,6,7,8,9]])

>>> np.hstack([a,b])
Traceback (most recent call last):
ValueError: ...
```

Write code below to fix the problem such that we get the output:

```
>>> np.hstack([a,b])
array([[10, 11, 12, 13, 14,  0,  1,  2,  3,  4],
       [15, 16, 17, 18, 19,  5,  6,  7,  8,  9]])
```

The problem was in the array **a.** Array **b** was already 2-D according to the required output. The problem can be fixed by

```
a = a.reshape(2,-1)
```

Q3. If we have the following array

```
>>> a = np.array([1,2,3])
```

What results will the following two code snippets return?

`>>> np.tile(a,5)`	`>>> np.repeat(a,5)`
`array([1, 2, 3, 1, 2, 3, 1, 2, 3, 1, 2, 3, 1, 2, 3])`	`array([1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 3, 3, 3, 3, 3])`

Q4. For the given array

```
>>> arr = np.arange(9).reshape(3,3)
```

Write the code to swap the middle and last columns and the resulting output array after the operation:

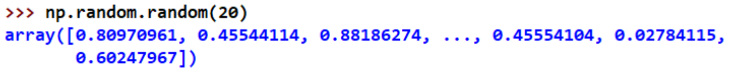
Code:	Output Array:

`arr = arr[:,[0,2,1]]`	`[[0, 2, 1],`
	` [3, 5, 4],`
	` [6, 8, 7]])`

Name:________________________________ Roll No:_______________ Section:____________________

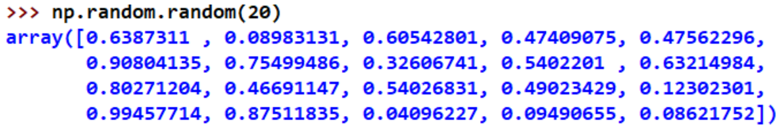
Q5. When numpy does not display full arrays like below:

From:



```
>>> np.random.random(20)
array([0.80970961, 0.45544114, 0.88186274, ..., 0.45554104, 0.02784115,
       0.60247967])
```

To:

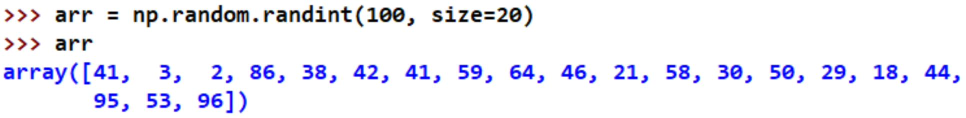


```
>>> np.random.random(20)
array([0.6387311 , 0.08983131, 0.60542801, 0.47409075, 0.47562296,
       0.90804135, 0.75499486, 0.32606741, 0.5402201 , 0.63214984,
       0.80271204, 0.46691147, 0.54026831, 0.49023429, 0.12302301,
       0.99457714, 0.87511835, 0.04096227, 0.09490655, 0.08621752])
```

Which method can help us display all 20 elements of the array?

a. numpy.set_display(threshold=20)
b. numpy.set_printoptions(threshold=20)
c. numpy.set_printoptions(precision=20)
d. numpy.set_display(arraysize=20)

Q6. Write a code snippet using numpy to extract numbers that are in the range between 55 and 90:



```
>>> arr = np.random.randint(100, size=20)
>>> arr
array([41,  3,  2, 86, 38, 42, 41, 59, 64, 46, 21, 58, 30, 50, 29, 18, 44,
       95, 53, 96])
```

```
Any of these:
  # Careful with the parenthesis here
   1. ((arr>=55) & (arr<=90))
   2. np.where((arr>=55) & (arr<=90))
   3. np.where(np.logical_and(arr>=55, arr<=90))
```

Q7. [FILL THE BLANK] We have the following two randomly initialized arrays. Fill in the blanks below with a numpy method to return the array at the end (i.e array([5, 6, 7])).

```
>>> a = np.random.randint(low=5, high=10, size=10)
>>> b = np.random.randint(low=8, high=13, size=10)
>>> a
array([6, 6, 7, 5, 9, 5, 6, 5, 8, 6])
>>> b
array([ 9,  9,  9,  8, 11, 10, 11, 12,  8, 10])

>>> np.setdiff1d(a,b)
array([5, 6, 7])
```

Q8. Use one-line code using numpy methods to print whether the median and mean of the following array are the same:

```
>>> arr = np.arange(81).reshape(9,9)
```

```
np.mean(arr) == np.median(arr)
```

Q9. The Z-score of data is calculated using the formula:

$$z = \frac{x - \mu}{\sigma}$$

$\mu =$ Mean

$\sigma =$ Standard Deviation

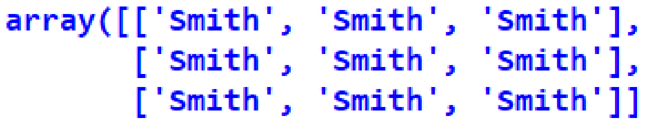
Write a Python function that takes a numpy array as input and returns the z-score values of the input data using numpy methods. (Do not use loops!)

```
def zscore(x):
        # x is a 2d numpy array taken as input
        # np.mean(x) and np.std(x) will return single values
        # However, x - np.mean(x) will return an array with the same dimensions of x
        # So will dividing by np.std(x)
         return (x - np.mean(x))/np.std(x)
```

Name:________________________________ Roll No:________________ Section:____________________

Q10. Use a numpy method to create an array of size 3x3 with all the values containing only your name:

For example:



```
array([['Smith', 'Smith', 'Smith'],
       ['Smith', 'Smith', 'Smith'],
       ['Smith', 'Smith', 'Smith']]
```

```
np.full([3,3],'Junaid')
```