Python Advance Assignment 1

1. What makes NumPy.shape() different from NumPy.size()?

NumPy.shape() is shape attribute of NumPy ndarray representing length of each dimension of array as a tuple.

Egs.

```
np.shape(np.eye(3)) => (3, 3)
np.shape([[1, 3]]) => (1, 2)
np.shape([0]) => (1,)
np.shape(0) => ()
```

NumPy.size() is the size attribute of numpy.ndarray representing total number of elements of the NumPy array as a integer value.

```
x = np.zeros((3, 5, 2), dtype=np.complex128)
x.size => 30
```

2. In NumPy, describe the idea of broadcasting.

Broadcasting is method that NumPy uses to allow array arithmetic between arrays with a different shape or size. Broadcasting solves the problem of arithmetic between arrays of differing shapes by in effect replicating the smaller array along the last mismatched dimension. So, during arithmetic operations with different shapes, subject to certain constraints, the smaller array is “broadcast” across the larger array so that they have compatible shapes.

Eg. Scalar and One/Two-Dimensional Array -

```
from numpy import array
a = array([1, 2, 3])
b = 2
a + b => [3 4 5]
# one-dimensional and two-dimensional
A = array([[1, 2, 3], [1, 2, 3]])
b = array([1, 2, 3])
A + b => [[2 4 6]  [2 4 6]]
```

3. What makes Python better than other libraries for numerical computation?

Python is open source, interpreted, interactive, object-oriented programming language.

Python lends itself to implementing complex or specialized algorithms for solving computational problems.

Python has built-in support for scientific computing. It has packages like Numpy for numerical computation, Scikits for multiple independent toolkits, Matplotlib for 2D plotting visualization.

Python is a highly extensible language ie if needed, we can write some of Python code in other languages like C++, means it can extended to other languages. This improves performance of the application.

Python has strong support for task automation: Python's built-in scripting features and multiple packages have strong support for task automation. Automation of repetitive tasks and performing data logging are easy and takes little effort.

Python can use a web front end. Python packages such as Django and Flask make it possible to develop and use Python as an API with a web front end. This functionality is particularly useful when using a cloud-based infrastructure as a platform to access high-performance computing (HPC) back ends.

Instead of old approaches like mathematical modeling and simulation, scientists have developed new data-driven, discovery-based, quantum computing approaches to solve scientific problems at scale. Many data-driven scientific problems use statistical or Bayesian analysis tools to solve specific classes of problems & Python has become the best programming language for data scientists. Cognitive approaches such as Machine learning and deep learning frameworks look for patterns and discover correlations are mostly Python-based. The quantum computing has opened entirely new problem-solving approaches & Python implementations allow communication between a scientific research application and the quantum computing system back-end.

4. How does NumPy deal with files?

NumPy provides functions to read and write data to files using the load() and save() methods, which support various file formats such as .txt, .csv, .npy, and .npz. These methods allow users to efficiently store and manipulate large numerical datasets in NumPy arrays.

5. Mention the importance of NumPy.empty().

The main use of NumPy empty is that it enables you to quickly create an array with a specific size and shape, without initializing entries The main advantage of np.empty over np.ones and np.zeros is that it's a little bit faster, particularly for large arrays.