Assignment 3

Question 1:

In [3]:
```
# Create a numpy array starting from 2 till 50 with a stepsize of 3.
import numpy as np
my_array = np.arange(2, 51, 3)
# printing of an array created by arange function
print(my_array)
```

```
[ 2  5  8 11 14 17 20 23 26 29 32 35 38 41 44 47 50]
```

Question 2 :

In [4]:
```
import numpy as np
my_list_one = []
my_list_two = []
for i in range(5):
    my_input = int(input())
    my_list_one.append(my_input)
print("List one :",my_list_one)
for i in range(5):
    my_input = int(input())
    my_list_two.append(my_input)
print("List Two", my_list_two)

# conversion of List to Array

my_array_one = np.array(my_list_one)
my_array_two = np.array(my_list_two)
print("List to array convert 1 :", my_array_one)
print("List to array convert 2 :", my_array_two)

# concatenation of arrays

array_concat = np.concatenate((my_array_one, my_array_two))
print("concatenation of arrays are :", array_concat)

# Sorting of array

print("Sorted array :", np.sort(array_concat))
```

```
1
2
3
4
5
List one : [1, 2, 3, 4, 5]
6
7
8
9
5
List Two [6, 7, 8, 9, 5]
List to array convert 1 : [1 2 3 4 5]
List to array convert 2 : [6 7 8 9 5]
concatenation of arrays are : [1 2 3 4 5 6 7 8 9 5]
Sorted array : [1 2 3 4 5 5 6 7 8 9]
```

Question 3:

In [7]:
```
import numpy as np
my_array_one = np.array([[1, 4, 7], [3, 6, 9], [1, 4, 7], [3, 6, 9]])
print("calculating Dimensions of a nd array is :", my_array_one.ndim)
print("Size of an array is :", my_array_one.size)
```

```
calculating Dimensions of a nd array is : 2
Size of an array is : 12
```

Question 4:

In [6]:
```
import numpy as np
arr = np.arange(int(input()))
print("Test 1D array (shape) :", arr.shape)
two_d_array_row = arr[np.newaxis]
print("Test` 2D array for row is :", two_d_array_row.shape)
two_d_array_col = arr[:, np.newaxis]
print("Test` 2D array for column is :", two_d_array_col.shape)
```

```
18
Test 1D array (shape) : (18,)
Test` 2D array for row is : (1, 18)
Test` 2D array for column is : (18, 1)
```

Question 5:

In [10]:
```
import numpy as np
my_array_one = np.square([1, 4, 5, 8, 79, 6, 3, 5, 7, 5, 9, 6, 4, 5])  # square o
my_array_two = np.square([4, 5, 8, 6, 9, 7, 4, 5, 6])  # square of numpy array
print(" horizontally stacked :", np.hstack((my_array_one, my_array_two)))
print("vertically stacked : ", np.hstack((my_array_one, my_array_two)))
```

```
 horizontally stacked : [   1   16   25   64 6241   36    9   25   49   25   81
36   16   25
   16   25   64   36   81   49   16   25   36]
vertically stacked :  [   1   16   25   64 6241   36    9   25   49   25   81
36   16   25
   16   25   64   36   81   49   16   25   36]
```

Question 6:

In [11]:
```
import numpy as np
arr = np.array([6, 8, 2, 4, 6, 1, 5, 9, 7])
unique, counts = np.unique(arr, return_counts=True)
arr = np.asarray((unique, counts)).T
print(arr)
```

```
[[1 1]
 [2 1]
 [4 1]
 [5 1]
 [6 2]
 [7 1]
 [8 1]
 [9 1]]
```