

```
In [ ]: 1 # Problem Statement 1:
        2 # Write a function so that the columns of the output matrix are powers of the
        3 # The order of the powers is determined by the increasing boolean argument. S
        4 # the i-th output column is the input vector raised element-wise to the power
        5 # HINT: Such a matrix with a geometric progression in each row is named for A
```

```
In [22]: 1 #creating a matrix with output columns based on input vector(N)
        2 import numpy as np
        3 x=np.array([1,2,3,5,7])
        4 N=5
        5 matrix = np.column_stack([x**(N-1-i) for i in range(N)])
        6 print(matrix)
```

```
[[ 1  1  1  1  1]
 [ 16  8  4  2  1]
 [ 81 27  9  3  1]
 [ 625 125 25  5  1]
 [2401 343 49  7  1]]
```

```
In [23]: 1 #Similar matrix is created using np.vander function in numpy
        2 import numpy as np
        3 x=np.array([1,2,3,5,7])
        4 N=5
        5 matrix1 = np.vander(x,N,increasing=False)
        6 print(matrix1)
```

```
[[ 1  1  1  1  1]
 [ 16  8  4  2  1]
 [ 81 27  9  3  1]
 [ 625 125 25  5  1]
 [2401 343 49  7  1]]
```

```
In [24]: 1 np.linalg.det(matrix)
```

```
Out[24]: 11519.999999999965
```

```
In [25]: 1 np.linalg.det(matrix1)
```

```
Out[25]: 11519.999999999965
```