Q1

In [1]:
```
import numpy as np
```

In [2]:
```
a = np.random.randint(99, size=(3,4))
a
```

Out[2]:
```
array([[48, 62, 37, 51],
       [53,  5, 61, 51],
       [45, 38, 68,  3]])
```

In [3]:
```
b = np.random.randint(99, size=(5,4))
b
```

Out[3]:
```
array([[44,  0, 28,  3],
       [33, 13, 57, 44],
       [41, 77, 43, 73],
       [20, 95, 31, 52],
       [ 5, 77, 38, 84]])
```

In [4]:
```
c = np.vstack([a,b])
c
```

Out[4]:
```
array([[48, 62, 37, 51],
       [53,  5, 61, 51],
       [45, 38, 68,  3],
       [44,  0, 28,  3],
       [33, 13, 57, 44],
       [41, 77, 43, 73],
       [20, 95, 31, 52],
       [ 5, 77, 38, 84]])
```

In [5]:
```
c.shape
```

Out[5]:
```
(8, 4)
```

Q2

In [6]:
```
def right_push(arr, push, num):
    arr = np.roll(arr,push)
    for i in range(push):
        arr[i] = num
    return arr
```

In [7]:
```
a = np.arange(1,11)
a
```

Out[7]:
```
array([ 1,  2,  3,  4,  5,  6,  7,  8,  9, 10])
```

In [8]:
```
b = a
```

```
b = right_push(b,1,0)
b
```

Out[8]:
```
array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9])
```

In [9]:
```
a.size==b.size
```

Out[9]:
```
True
```

In [10]:
```
print("a_shape = "+str(a.shape))
print("b_shape = "+str(b.shape))
```

```
a_shape = (10,)
b_shape = (10,)
```

Q3

In [11]:
```
def create_mask_add(arr, mask):
    b = np.tril(arr)
    np.fill_diagonal(b,0)

    return (b - arr)
```

In [12]:
```
x = np.random.randint(1,10,size = (3,3))
x
```

Out[12]:
```
array([[6, 7, 4],
       [2, 3, 3],
       [7, 2, 6]])
```

In [13]:
```
mask = create_mask_add(x, 0)
mask
```

Out[13]:
```
array([[-6, -7, -4],
       [ 0, -3, -3],
       [ 0,  0, -6]])
```

In [14]:
```
new_x = x + mask
new_x
```

Out[14]:
```
array([[0, 0, 0],
       [2, 0, 0],
       [7, 2, 0]])
```

In [15]:
```
mask_mul = np.array([[0,0,0],[1,0,0],[1,1,0]])
mask_mul
```

Out[15]:
```
array([[0, 0, 0],
       [1, 0, 0],
       [1, 1, 0]])
```

In [16]:
```
new_x = x * mask_mul
new_x
```

Out[16]:
```
array([[0, 0, 0],
       [2, 0, 0],
       [7, 2, 0]])
```

Q4

In [17]:
```
a = np.full((30,30), 2, dtype=int)
```

In [18]:
```
b = np.full((30,30), 3, dtype=int)
```

In [19]:
```
c = np.full((30,30), 4, dtype=int)
```

In [20]:
```
d = np.full((30,30), 6, dtype=int)
```

In [21]:
```
d.shape
```

Out[21]:
```
(30, 30)
```

In [22]:
```
average = (a+b+c+d)/4
average
```

Out[22]:
```
array([[3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75
,
        3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75
,
        3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75],
       [3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75
,
        3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75
,
        3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75],
       [3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75
,
        3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75
,
        3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75],
       [3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75
,
        3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75
,
        3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75],
       [3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75
,
        3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75
,
        3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75],
       [3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75
,
```

```
       ,
              3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75
       ,
              3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75, 3.75]])
```

In [23]:
```
avg = np.mean([a,b,c,d])
avg
```

Out[23]:
```
3.75
```