In [1]:

```
# Load libraries
import numpy as np
```

In [2]:

```
# Create a random array with given dimension
array = np.random.rand(6,4)
# Print ot for checking
print(array)
```

```
[[0.74614954 0.34325835 0.41055068 0.9669944 ]
 [0.37868342 0.64385264 0.93182578 0.97856728]
 [0.29643537 0.41870843 0.39535716 0.24483783]
 [0.88945189 0.53854462 0.49863201 0.64492517]
 [0.2772637  0.84601776 0.92242692 0.94095503]
 [0.4035588  0.28487034 0.15173798 0.07102357]]
```

In [3]:

```
# Define a function that takes an array as input
# And returns it's dimension
def check_true(array,y):
    for i in range(y):
        if array[i]==False:
            return False
    return True

def dimension(array):
    x,y=np.shape(array)
    x_new=x
    y_new=y
    for i in range(x):
        l=array[i]==array[i][0]
        if check_true(l,y):
            x_new-=1
    Trans = [[array[j][i] for j in range(len(array))] for i in range(len(array[0]))]
    for i in range(y):
        l=Trans[i]==Trans[i][0]
        if check_true(l,x):
            y_new-=1
    if x_new==0:
        x_new+=1
    if y_new==0:
        y_new+=1
    print((x_new,y_new))
    if x_new!=1 and y_new!=1:
        return 2
    elif x_new!=1 or y_new!=1:
        return 1
    return 0
```

In [4]:

```
# Test case 1 ---  array(0,1)
# For doing this Lets first replace first row by its second element
# Then make every column equal to it's first element
# In this way everything in matrix will become array(0,1)
array = np.random.rand(6,4)
array[0]=array[0][1]
x,y=np.shape(array)
for i in range(y):
    for j in range(x):
        array[j][i]=array[0][i]
print(dimension(array))
```

```
(1, 1)
0
```

In [5]:

```
# Test case 2 ---  array(4,3)
# For doing this Lets first replace fifth row by its fourth element
# Then make every column equal to it's fifth element
# In this way everything in matrix will become array(4,3)
array = np.random.rand(6,4)
array[4]=array[4][3]
x,y=np.shape(array)
for i in range(y):
    for j in range(x):
        array[j][i]=array[4][i]
print(dimension(array))
```

```
(1, 1)
0
```

In [6]:

```
# Test case 3 ---  array(:,1)
# This indicates we want all rows of second column
# For doing this Lets replace every column except the second one by it's average
# So we will have only 2nd column now
array = np.random.rand(6,4)
x,y=np.shape(array)
for i in range(y):
    if i!=1:
        avg=0
        for j in range(x):
            avg+=array[j][i]
        avg/=x
        for j in range(x):
            array[j][i]=avg
print(dimension(array))
```

```
(6, 1)
1
```

In [7]:

```
# Test case 4 ---  array(1,:)
# This indicates we want all columns of second row
# For doing this Lets replace every row except the second one by it's maximum
# So we will have only 2nd row now
array = np.random.rand(6,4)
x,y=np.shape(array)
for i in range(x):
    if i!=1:
        max=0
        for j in range(y):
            if max < array[i][j]:
                max = array[i][j]
        array[i]=max
print(dimension(array))
```

```
(1, 4)
1
```

In [8]:

```
# Test case 5 ---  array(3,:)
# This indicates we want all columns of fourth row
# For doing this Lets replace every row except the fourth one by it's minimum
# So we will have only 4th row now
array = np.random.rand(6,4)
x,y=np.shape(array)
for i in range(x):
    if i!=3:
        min=1
        for j in range(y):
            if min > array[i][j]:
                min = array[i][j]
        array[i]=min
print(dimension(array))
```

```
(1, 4)
1
```

In [9]:

```
# Test case 6 ---  array(1:4,:)
# This indicates we want all columns of second to fourth rows
# For doing this Lets replace every row except the second to fourth one by it's minimum
# So we will have only 2nd to 4th rows now
array = np.random.rand(6,4)
x,y=np.shape(array)
for i in range(x):
    if i not in range(1,4):
        min=1
        for j in range(y):
            if min > array[i][j]:
                min = array[i][j]
        array[i]=min
print(dimension(array))
```

```
(3, 4)
2
```

In [10]:

```
# Test case 7 ---  array(:,0:3)
# This indicates we want all rows of first to third column
# For doing this Lets replace every column except the first to third one by it's sum
# So we will have only 1st to 3rd column now
array = np.random.rand(6,4)
x,y=np.shape(array)
for i in range(y):
    if i not in range(0,3):
        sum=0
        for j in range(x):
            sum+=array[j][i]
        for j in range(x):
            array[j][i]=sum
print(dimension(array))
```

```
(6, 3)
2
```

In [ ]: