



THỰC NGHIỆM SỬ DỤNG CÔNG CỤ

- **Numpy (Numeric Python):** là một thư viện toán học phổ biến và mạnh mẽ của Python. Cho phép làm việc hiệu quả với ma trận và mảng, đặc biệt là dữ liệu ma trận và mảng lớn với tốc độ xử lý nhanh hơn nhiều lần khi chỉ sử dụng “core Python” đơn thuần.
- **Cài đặt thư viện Numpy:** Mở Command Prompt và gõ lệnh

```
pip install numpy
```

- **Khai báo thư viện:**

```
import numpy as np
```

- Khởi tạo mảng một chiều

```
import numpy as np
#Khởi tạo mảng một chiều với kiểu dữ liệu các phần tử là Integer
arr = np.array([1,3,4,5,6], dtype = int)
#Khởi tạo mảng một chiều với kiểu dữ liệu mặc định
arr = np.array([1,3,4,5,6])
print(arr)
```

```
[1 3 4 5 6]
```

- Khởi tạo mảng hai chiều



```
import numpy as np  
arr1 = np.array([(4,5,6), (1,2,3)], dtype = int)  
print(arr1)
```



```
[[4 5 6]  
 [1 2 3]]
```

Khởi tạo với các hàm có sẵn:

- **`np.zeros((3,4), dtype = int)`**: Tạo mảng hai chiều các phần tử 0 với kích thước 3x4.
- **`np.ones((2,3,4), dtype = int)`**: Tạo mảng 3 chiều các phần tử 1 với kích thước 2x3x4.
- **`np.arange(1,7,2)`**: Tạo mảng với các phần tử từ 1 - 6 với bước nhảy là 2.
- **`np.full((2,3),5)`**: Tạo mảng 2 chiều các phần tử 5 với kích thước 2x3.
- **`np.eye(4, dtype=int)`**: Tạo ma trận đơn vị với kích thước là 4x4.
- **`np.random.random((2,3))`**: Tạo ma trận các phần tử ngẫu nhiên với kích thước 2x3.

- **dtype**: Kiểu dữ liệu của phần tử trong mảng.
- **shape**: Kích thước của mảng.
- **size**: Số phần tử trong mảng.
- **ndim**: Số chiều của mảng.



```
import numpy as np
arr = np.array([(4,5,6), (1,2,3)], dtype = int)
print("Kiểu dữ liệu của phần tử trong mảng:", arr.dtype)
print("Kích thước của mảng:", arr.shape)
print("Số phần tử trong mảng:", arr.size)
print("Số chiều của mảng:", arr.ndim)
```



```
Kiểu dữ liệu của phần tử trong mảng: int64
Kích thước của mảng: (2, 3)
Số phần tử trong mảng: 6
Số chiều của mảng: 2
```

- **dtype**: Kiểu dữ liệu của phần tử trong mảng.
- **shape**: Kích thước của mảng.
- **size**: Số phần tử trong mảng.
- **ndim**: Số chiều của mảng.



```
import numpy as np
arr = np.array([(4,5,6), (1,2,3)], dtype = int)
print("Kiểu dữ liệu của phần tử trong mảng:", arr.dtype)
print("Kích thước của mảng:", arr.shape)
print("Số phần tử trong mảng:", arr.size)
print("Số chiều của mảng:", arr.ndim)
```



```
Kiểu dữ liệu của phần tử trong mảng: int64
Kích thước của mảng: (2, 3)
Số phần tử trong mảng: 6
Số chiều của mảng: 2
```

- Các phần tử trong mảng được đánh số từ 0 trở đi
- **arr[i]**: Truy cập tới phần tử thứ i của mảng 1 chiều.
- **arr1[i,j]**: Truy cập tới phần tử hàng i, cột j của mảng 2 chiều.
- **arr2[n,i,j]**: Truy cập tới phần tử chiều n, hàng i, cột j của mảng 3 chiều.
- **arr[a:b]**: Truy cập tới các phần tử từ a đến b-1 trong mảng 1 chiều.
- **arr1[:,i]**: Truy cập tới phần tử từ cột 0 đến cột i-1, của tất cả các hàng trong mảng 2 chiều.



```
import numpy as np
arr = np.array([1,3,4,5,6], dtype = int)# mảng 1 chiều
arr1 = np.array([(4,5,6), (1,2,3)], dtype = int)#mảng 2 chiều
arr2 = np.array([(2,4,0,6), (4,7,5,6)],
                [(0,3,2,1), (9,4,5,6)],
                [(5,8,6,4), (1,4,6,8)]), dtype = int)#mảng 3 chiều
print("arr[2]=", arr[2])
print("arr1[1:2]=", arr1[1,2])
print("arr2[1,2,3]=", arr2[1,1,3])
print("arr[0:3]=", arr[0:3])
print("arr1[:, :1]=", arr1[:, :2])
```

```
arr[2]= 4
arr1[1:2]= 3
arr2[1,2,3]= 6
arr[0:3]= [1 3 4]
arr1[:, :1]= [[4 5]
              [1 2]]
```

- **arr.max()** hoặc **np.max(arr)**: Lấy giá trị lớn nhất của mảng arr.
- **arr.min()** hoặc **np.min(arr)**: Lấy giá trị nhỏ nhất của mảng arr.
- **arr.sum()** hoặc **np.sum(arr)**: Tổng tất cả các phần tử trong mảng arr.
- **arr.mean()** hoặc **np.mean(arr)**: Trung bình cộng của tất cả các phần tử trong mảng arr.
- **np.median(arr)**: Trả về giá trị trung vị của mảng arr.



```
import numpy as np
arr = np.array([1,3,4,5,6], dtype = int)# mảng 1 chiều
print("Giá trị lớn nhất của mảng arr là:", np.max(arr))
print("Giá trị nhỏ nhất của mảng arr là:", np.min(arr))
print("Tổng tất cả các phần tử của mảng arr là:", np.sum(arr))
print("Trung bình cộng tất cả các phần tử của mảng arr là:", np.mean(arr))
print("Giá trị trung vị của mảng arr là:", np.median(arr))
```

Giá trị lớn nhất của mảng arr là: 6

Giá trị nhỏ nhất của mảng arr là: 1

Tổng tất cả các phần tử của mảng arr là: 19

Trung bình cộng tất cả các phần tử của mảng arr là: 3.8

Giá trị trung vị của mảng arr là: 4.0

```
a = np.array([2,1,3,4,5])
```

Toán tử	Ví dụ	Kết quả
(+) Một số với mảng	$3 + \text{arr}$ $\text{arr} + 3$	[4, 6, 7, 8, 9]
(+) Mảng với mảng	$\text{arr} + a$ $a + \text{arr}$	[3, 4, 7, 9, 11]
(-) Một số với mảng	$\text{arr} - 3$ $3 - \text{arr}$	[-2, 0, 1, 2, 3] [2, 0, -1, -2, -3]

```
a = np.array([2,1,3,4,5])
```

(-) Mảng với mảng	arr - a	[-1, 2, 1, 1, 1]
	a - arr	[1, -2, -1, -1, -1]
(*) Một số với mảng	arr * 3	[3, 9, 12, 15, 18]
	3 * arr	
(*) Mảng với mảng	arr * a	[2, 3, 12, 20, 30]
	a * arr	
(/) Một số với mảng	arr / 3	[0.33333333, 1., 1.33333333, 1.66666667, 2.]
	3 / arr	[3., 1., 0.75, 0.6, 0.5]

```
a = np.array([2,1,3,4,5])
```

(/) Mảng với mảng	arr / a	[0.5, 3., 1.33333333, 1.25, 1.2]
	a / arr	[2., 0.33333333, 0.75, 0.8, 0.83333333]
Hạng của ma trận	np.rank(arr)	1
	np.rank(arr1)	2

Bài tập 1: Viết chương trình Python nhập một mảng hai chiều các số thực A (m hàng, n cột) từ bàn phím.

- Tìm giá trị lớn nhất và nhỏ nhất trên mỗi cột
- Tìm phần tử lớn nhất và phần tử nhỏ nhất của mảng A cùng các chỉ số hàng và cột của 2 phần tử này.
- Trong mảng A có bao nhiêu phần tử bằng phần tử lớn nhất.

Gợi ý:

- Sử dụng thư viện **numpy** để khởi tạo mảng A và thực hiện các phép toán trên mảng.
- Câu lệnh **np.amax(A, axis=0)** trả về giá trị lớn nhất trên mỗi cột của mảng A, còn **np.amin(A, axis=0)** trả về giá trị nhỏ nhất trên mỗi cột.
- Để tìm phần tử lớn nhất và nhỏ nhất của mảng A cùng với chỉ số hàng và cột của 2 phần tử này, chúng ta sử dụng **np.amax(A)** và **np.amin(A)** để tìm giá trị lớn nhất và nhỏ nhất của mảng A, sau đó sử dụng hàm **np.unravel_index** để chuyển đổi chỉ số dạng 1 chiều thành chỉ số dạng 2 chiều.