

# CẤU TRÚC DỮ LIỆU & GIẢI THUẬT NÂNG CAO

CHỦ ĐỀ: Mạng động nhiều chiều

Giảng viên hướng dẫn: Nguyễn Thanh Sơn

Thành viên nhóm:

Nguyễn Vũ Dương – 20520465

Võ Nguyễn Hoài Nam – 20520645

Phạm Phước An - 20520375

# Nội Dung



GIỚI  
THIỆU

CƠ SỞ  
LÝ  
THUYẾT

CÀI ĐẶT  
THAO TÁC

ỨNG  
DỤNG

QUIZ





# 1. GIỚI THIỆU



## MỤC TIÊU CHỦ ĐỀ

Tìm hiểu rõ về các thao tác liên quan đến mảng động nhiều chiều trong toán học.

## PHƯƠNG PHÁP NGHIÊN CỨU

Nghiên cứu các hàm trong thư viện numpy.

## ĐỐI TƯỢNG VÀ PHẠM VI NGHIÊN CỨU

Nghiên cứu về cách thức hoạt động của các hàm trong thư viện numpy sau đó tự code ra các hàm đó thao tác trên kiểu dữ liệu list.



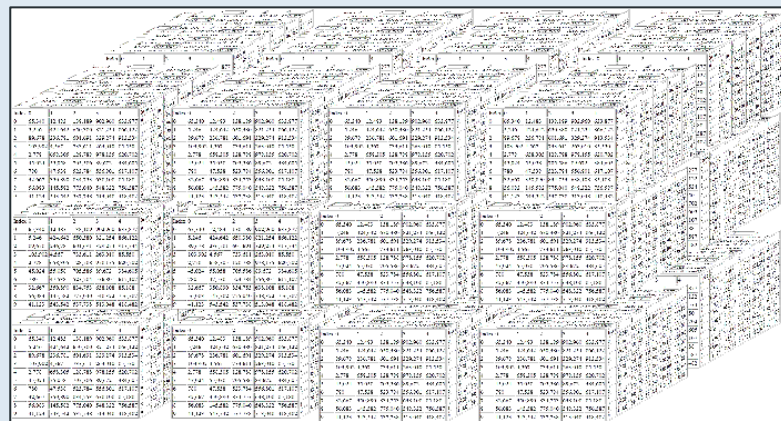
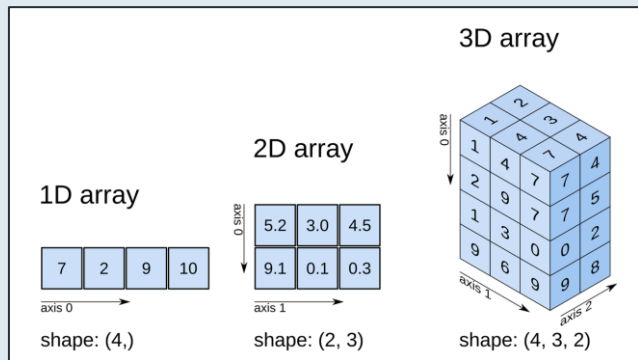


## 2. CƠ SỞ LÝ THUYẾT



## 2.1 Khái niệm về mảng động nhiều chiều

- Mảng động nhiều chiều là là một container nhiều chiều gồm các items có cùng loại và kích thước.
- Số lượng chiều và items trong mảng được định nghĩa bởi shape (hình dạng) của nó.





## 2.2 Shape (hình dạng) của mảng



### a. Shape trong Numpy

- Biểu diễn dưới dạng bộ dữ liệu (tuple) gồm kích thước mỗi chiều của mảng.
- `Numpy.shape` thường được dùng để lấy hình dạng hiện tại của mảng, nhưng cũng có thể thay đổi lại hình dạng mảng.

## 2.2 Shape (hình dạng) của mảng

### b. Shape trên list

- Shape trên list không có hàm cho trước, do người lập trình cài đặt.
- Ý tưởng tìm shape:

$$S = ()$$

$$T = \left[ \left[ [1,2], [3,4] \right], \left[ [5,6], [7,8] \right] \right] \rightarrow \text{len}(T) = 2 \rightarrow S = (2,)$$

$$T[0] = \left[ [1,2], [3,4] \right] \rightarrow \text{len}(T[0]) = 2 \rightarrow S = (2,2)$$

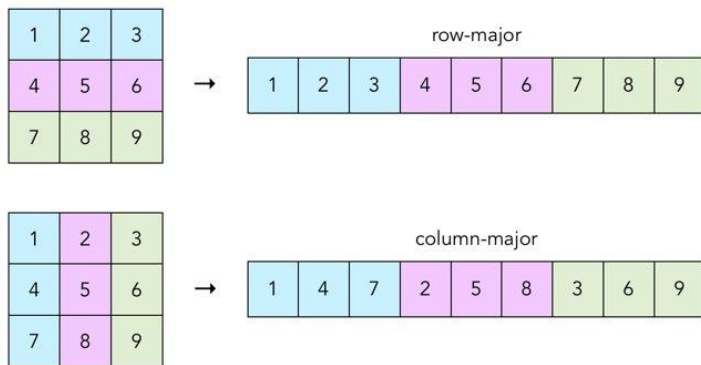
$$T[0][0] = [1,2] \rightarrow \text{len}(T[0][0]) = 2 \rightarrow S = (2,2,2)$$



## 2.2 Shape (hình dạng) của mảng

### c. Reshape

- Reshape trong numpy dùng để thay đổi hình dạng của mảng theo Fortran hoặc C-like order.
- Ví dụ:





## 2.2 Shape (hình dạng) của mảng



### c. Reshape

- Ý tưởng tìm reshape không dùng thư viện numpy:
- Giảm chiều của list về 1 chiều duy nhất
- Phân chia thành nhiều list con nằm trong 1 list mới.

Ví dụ:

$$A = \left[ \left[ [1,2], [3,4] \right], \left[ [5,6], [7,8] \right] \right] \rightarrow \left[ [1,2,3,4], [5,6,7,8] \right] \rightarrow [1,2,3,4,5,6,7,8]$$
$$\text{Reshape}(A, (4,2)) \rightarrow A = \left[ [1,2], [3,4], [5,6], [7,8] \right]$$



## 2.3 Resize

- Giảm chiều list về 1 chiều
- Có 3 trường hợp giữa  $S_1$  (tổng số phần tử list cũ) và  $S_2$  (tổng số phần tử list mới)
- $S_1 = S_2$  thì reshape
- $S_1 > S_2$  thì sao chép  $S_2$  phần tử từ list
- $S_1 < S_2$  thì sao chép  $2*S_2 - S_1$  phần tử từ list



## 2.3 Resize

- Trường hợp 1:

```
L = reshape(arange(0,18),(3,3,2))
```

```
D = resize(L,(3,3,2))
```

```
D
```

```
[[[ 0, 1], [ 2, 3], [ 4, 5]],
```

```
[[ 6, 7], [ 8, 9], [10, 11]],
```

```
[[12, 13], [14, 15], [16, 17]]]
```



## 2.3 Resize

- Trường hợp 2:

```
L = reshape(arange(0,18),(3,3,2))  
D = resize(L,(3,3,1))  
D  
[[[0], [1], [2]], [[3], [4], [5]], [[6], [7], [8]]]
```



## 2.3 Resize

- Trường hợp 3:

```
L = reshape(arange(0,18),(3,3,2))
D = resize(L,(3,3,3))
D
[[[ 0, 1, 2], [ 3, 4, 5], [ 6, 7, 8]],
 [[ 9, 10, 11], [12, 13, 14], [15, 16, 17]],
 [[ 0, 1, 2], [ 3, 4, 5], [ 6, 7, 8]]]
```

## 2.4 Chuyển vị của ma trận nhiều chiều

- Với ma trận 2 chiều, ta xét  $i, j$  lần lượt là dòng, cột của ma trận  
→  $A^T_{i,j} = A_{j,i}$
- Với ma trận 3 chiều, ta xét  $i, j, k$  lần lượt là chỉ số ở mỗi chiều của ma trận  
→  $A^T_{i,j,k} = A_{k,j,i}$
- Với ma trận  $n$  chiều tương tự như trên.

$A =$


1	56	72	83
45	22	65	0
88	7	-1	99

$A^T =$

1	45	88
56	22	7
72	65	-1
83	0	99



## 2.5 Ghép các mảng (Concatenating Arrays)

- Trong quá trình xử lý dữ liệu, rất nhiều bài toán đòi hỏi cần ghép nối nhiều mảng với nhau để mở rộng hàng hay cột.
  - Khi ghép các mảng đa chiều, chúng ta ghép các mảng theo trục.
- 





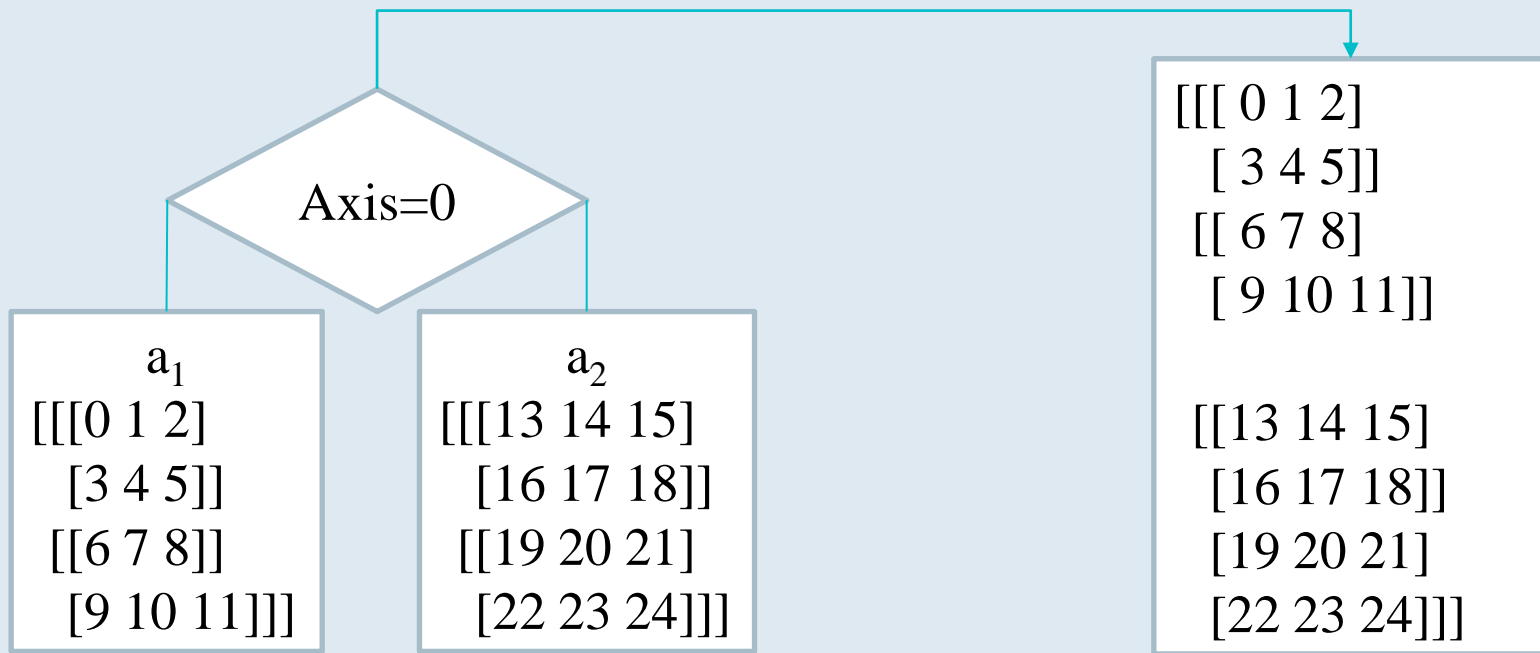
## 2.5 Ghép các mảng (Concatenating Arrays)

- Điều kiện ghép mảng là cùng size và cùng axis.
- Cú pháp `numpy.concatenate(a1,a2,...,axis)`.

## 2.5 Ghép các mảng (Concatenating Arrays)

- Ví dụ:

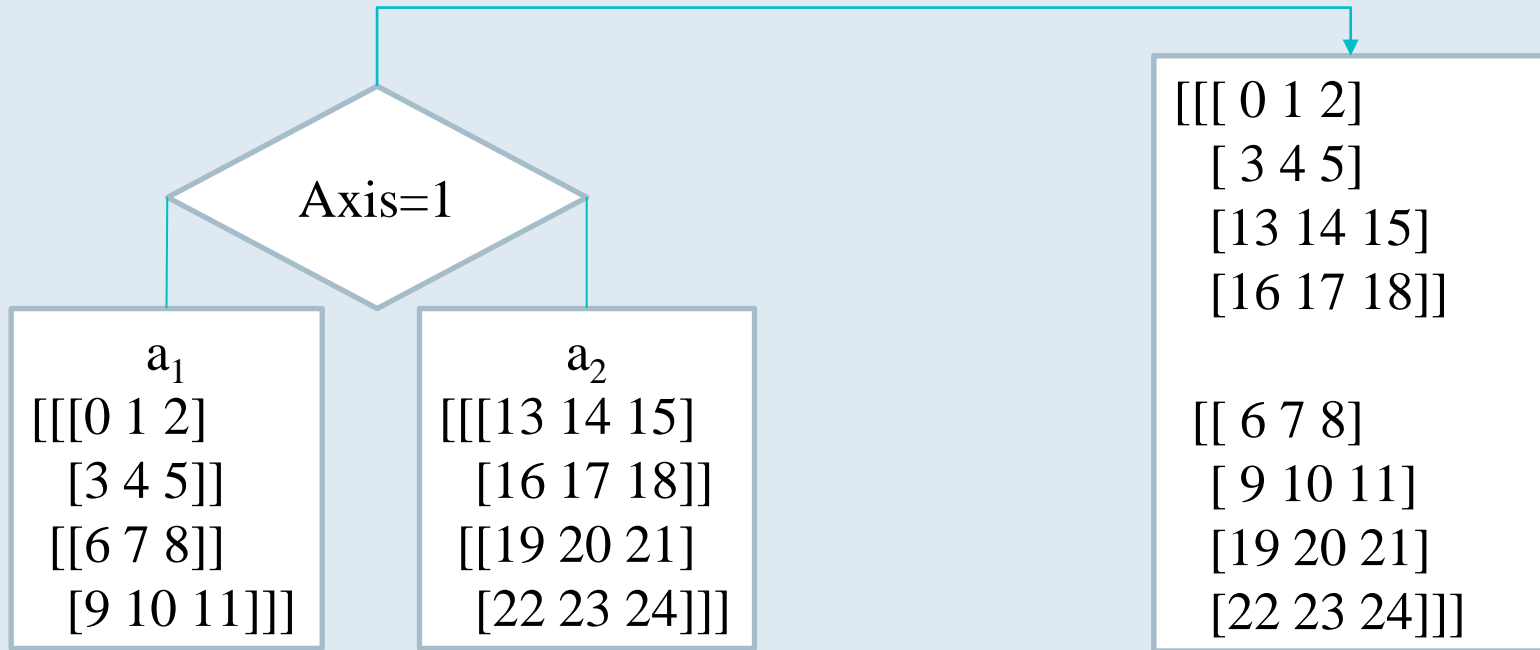
Concatenate



## 2.5 Ghép các mảng (Concatenating Arrays)

- Ví dụ:

Concatenate





## 2.6 Sự cần thiết của mảng động nhiều chiều

- Sử dụng mảng động nhiều chiều thì ta có thể resize kích thước của bức ảnh mà vẫn giữ được đặc trưng của bức ảnh.
- Việc resize các bức ảnh về cùng kích thước có ích cho các model (mô hình) trong machine learning hoặc deeplearning dễ dàng học hơn và tối ưu hơn.



### 3. CÀI ĐẶT THAO TÁC



## 4. ỨNG DỤNG





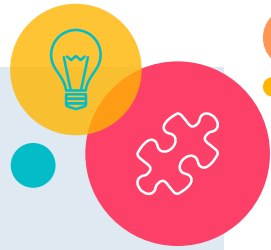
Minimum  
spanning trees

Image

Audio



## 4.1 Minimum spanning trees

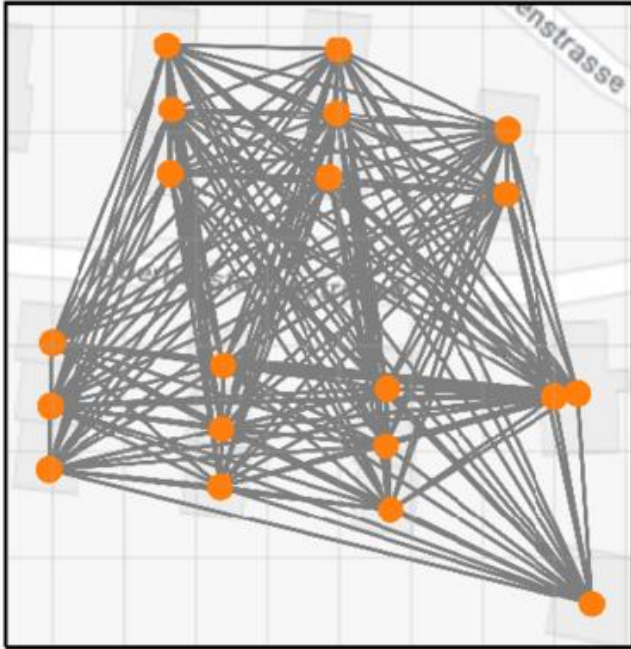


Cây khung nhỏ nhất được sử dụng:

- + Trong thiết kế mạng.
- + Theo dõi và xác minh khuôn mặt trong thời gian thực.
- + Các giao thức trong khoa học máy tính để tránh các chu kỳ mạng.
- + Phối màu



Fully Connected Network

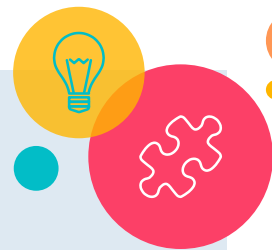


MST network





## 4.2 Image



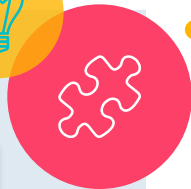
Ảnh số được lưu trong máy tính dưới dạng ma trận kích thước



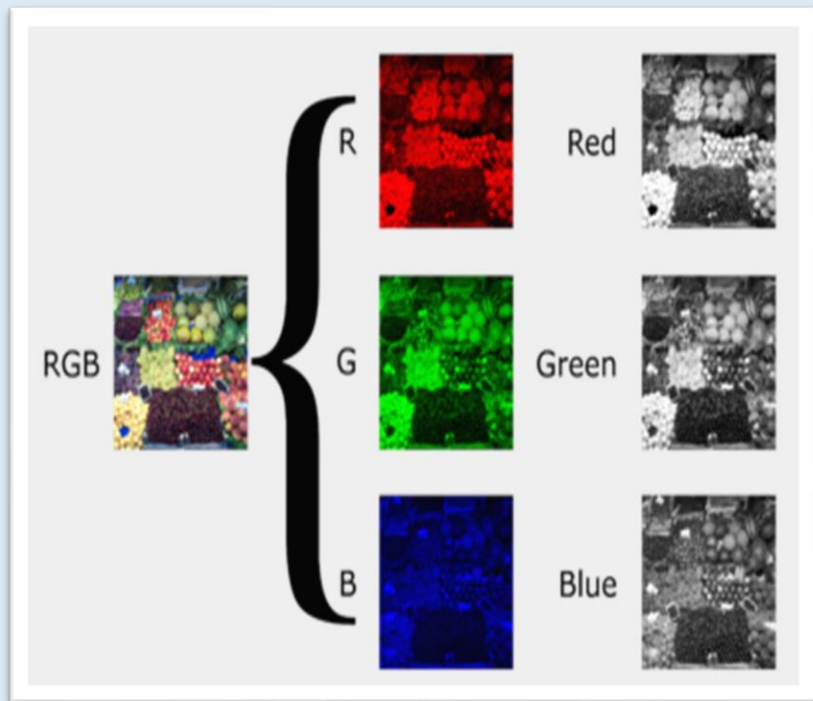
Colored Image  
( $m \times n \times 3$ )

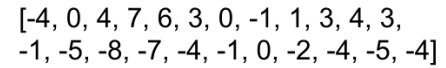


Gray Image  
( $m \times n \times 1$ )



0	2	15	0	0	11	10	0	0	0	9	9	0	0
0	0	0	4	60	157	236	255	255	177	95	61	32	0
0	10	16	119	238	255	244	245	243	250	249	255	222	103
0	14	170	255	255	244	254	255	253	245	255	249	253	251
2	98	255	228	255	251	254	211	141	116	122	215	251	238
13	217	243	255	155	33	226	52	2	0	10	13	232	255
16	229	252	254	49	12	0	0	7	7	0	70	237	252
6	141	245	255	212	25	11	9	3	0	115	236	243	255
0	87	252	250	248	215	60	0	1	121	252	255	248	144
0	13	113	255	255	245	255	182	181	248	252	242	208	36
1	0	5	117	251	255	241	255	247	255	241	162	17	0
0	0	0	4	58	251	255	246	254	253	255	120	11	0
0	0	4	97	255	255	255	248	252	255	244	255	182	10
0	22	206	252	246	251	241	100	24	113	255	245	255	194
0	111	255	242	255	158	24	0	0	6	39	255	232	230
0	218	251	250	137	7	11	0	0	0	2	62	255	250
0	173	255	255	101	9	20	0	13	3	13	182	251	245
0	107	251	241	255	230	98	55	19	118	217	248	253	255
0	18	146	250	255	247	255	255	255	249	255	240	255	129
0	0	23	113	215	255	250	248	255	255	248	248	118	14
0	0	6	1	0	52	153	233	255	252	147	37	0	0
0	0	5	5	0	0	0	0	0	14	1	0	6	6







## 5. QUIZ





# Thanks!

## Any questions?

You can find me at @username & user@mail.me

