

# Data Structure & Algorithm

## Data Structure

### Array

# 1. Định nghĩa

- Mảng là một tập hợp các phần tử có cùng một kiểu:

1	5	8	2	9
---	---	---	---	---

'B'	'R'	'O'	'W'	'N'
-----	-----	-----	-----	-----

## 2. Phân loại

- Mảng 1 chiều:

0	1	2	3	4
1	5	8	2	9



`array[1] = 5`

- Mảng nhiều chiều:

	0	1	2	3	4
0	1	5	8	2	9
1	0	3	8	0	8
2	5	8	4	9	7



`array[2][4] = 7`

## 2. Phân loại

### - Mảng tĩnh:

- Cần phải chỉ định số lượng phần tử khi khai báo.
- Được tạo trong vùng nhớ Stack.

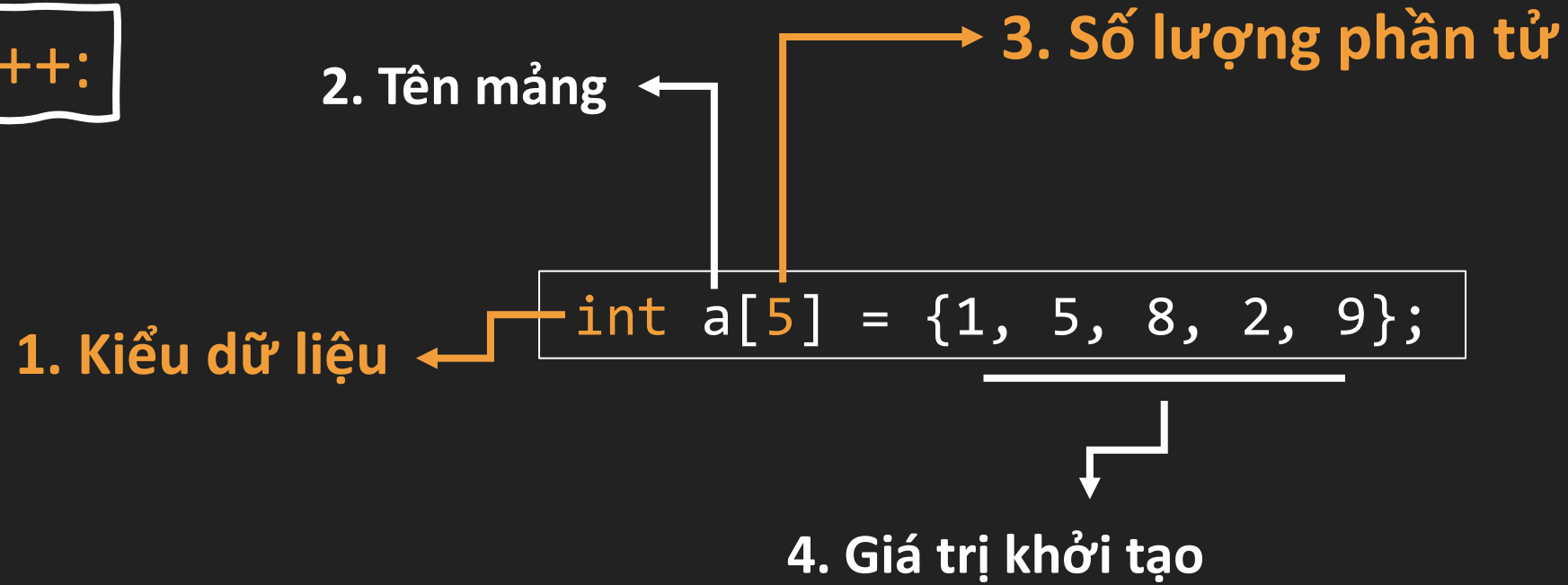
### - Mảng động:

- Không cần chỉ định số lượng phần tử khi khai báo.
- Được tạo trong vùng nhớ Heap.

# 3. Khai báo

## A- Mảng tĩnh

C/C++:



0	1	2	3	4
1	5	8	2	9

# 3. Khai báo

## A- Mảng tĩnh

**C/C++** : Các cách khai báo khác.

➤ `int a[5];`

➤ `int a[5] = {0};`

➤ `int a[] = {1, 5, 8, 2, 9};`

**Java** : N/A



0	1	2	3	4
0	0	0	0	0

0	1	2	3	4
1	5	8	2	9

# 3. Khai báo

## B- Mảng động

C:

```
int* a = malloc(sizeof(int)*5);
```

<stdlib.h>

```
free(a);
```

C++:

```
int* a = new int[5];
```

```
delete[] a;
```

Java:

```
int[] a = new int[5];
```

# 3. Khai báo

## C- Sử dụng thư viện

C++ STL:

```
vector<int> a;
```

```
vector<int> a(5, 0);
```

```
int n = a.size();
```

```
a.push_back(10);
```

```
a.resize(10, 1);
```

```
a.clear();
```

➤ More details about STL and vector :

The C++

STANDARD  
TEMPLATE  
LIBRARY

#01

VECTOR



# Data Structure & Algorithm

