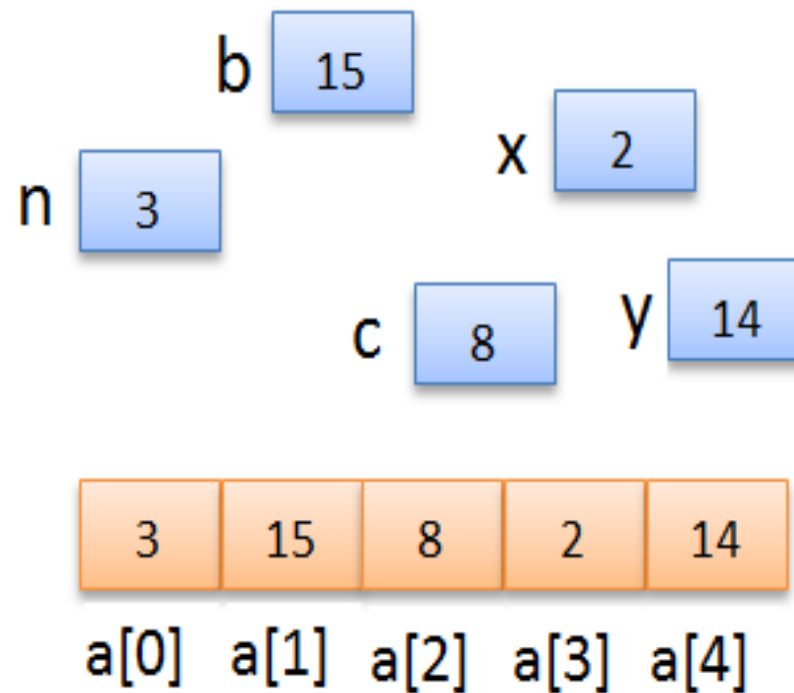


BÀI 7 MẢNG



Giảng viên: **Vũ Thương Huyền – Khoa Công nghệ thông tin – ĐH Thủy Lợi**

Email: **huyenvt@tlu.edu.vn**

NỘI DUNG

- Khái niệm về mảng
- Mảng một chiều
- Mảng hai chiều

KHÁI NIỆM VỀ MẢNG

- **Mảng** là tập hợp hữu hạn các phần tử có cùng kiểu dữ liệu
- Các phần tử lưu trữ ở các vị trí kế tiếp nhau trong bộ nhớ
- Kích thước của mảng bằng số phần tử
- Mảng một chiều, mảng nhiều chiều

0	1	2	3	4	5
6	23	12	9	14	52
0	0	0	0	0	0
x	x	x	x	x	x
0	0	0	0	0	0
0	0	0	0	0	0
1	1	1	1	1	1
2	3	4	5	6	7

MẢNG MỘT CHIỀU

▪ Khai báo: `kieudulieu tenmang [sophantu];`

- Phần tử của mảng truy nhập thông qua tên mảng
- Các phần tử được đánh số từ **0** → **(số phần tử - 1)**
- Số phần tử là hằng số

▪ Ví dụ:

```
int x[10];  
  
char hoten[50];  
  
float a[10], b[20];
```

MẢNG MỘT CHIỀU

- Khởi tạo giá trị khi khai báo:

```
kiAUDulieu tenmang [sophantu] = {giatri1, giatri2...., giatriN};
```



Chú ý:

- Các giá trị khởi tạo là **hằng số**
- Số lượng giá trị khởi tạo ít hơn số phần tử thì các phần tử còn lại sẽ có giá trị 0
- Có thể không cần khai báo kích thước của mảng khi khởi tạo

MẢNG MỘT CHIỀU

- Ví dụ 1: Khởi tạo cho mọi phần tử của mảng

```
char a[5] = {'A', 'B', 'C', 'D', 'E'};
```



- Ví dụ 2: Khởi tạo cho một số phần tử của mảng

```
int b[5] = {1, 2, 3};
```



NHẬP GIÁ TRỊ CHO CÁC PHẦN TỬ CỦA MẢNG

➤ Nhập trực tiếp thông qua chỉ số

```
b[0] = 1;
```

```
b[1] = 2;
```

➤ Nhập thông qua vòng lặp **for**

```
for(int i=0; i< n; i++)  
{  
    cout<<"a["<<i<<"]=";<<cin>>a[i];  
}
```



Chỉ số phải nhỏ hơn số phần tử của mảng

TRUY XUẤT GIÁ TRỊ CÁC PHẦN TỬ CỦA MẢNG

➤ Thông qua chỉ số

```
int a = b[0];  
cout<<b[1];
```

➤ Thông qua vòng lặp **for**

```
for(int i=0; i< n; i++)  
{  
    cout<<"a["<<i<<"]=";cout<<a[i]<<endl;  
}
```


MẢNG MỘT CHIỀU

- Ví dụ:

- Đọc vào dãy n số nguyên ($n \leq 100$)
- Hiển thị dãy số vừa nhập ra màn hình

MẢNG MỘT CHIỀU

▪ Ví dụ:

```
1  #include <iostream>
2  using namespace std;
3  int main()
4  {
5      int n, a[100];
6      cout<<"nhap so phan tu cua day n=";cin>>n;
7      for(int i=0;i<n;i++)
8      {
9          cout<<"a["<<i<<"]=";
10         cin>>a[i];
11     }
12     cout<<"Day so vua nhap"<<endl;
13     for(int i=0;i<n;i++)
14         cout<<a[i]<<" ";
15     return 0;
16 }
```



BÀI TẬP

▪ **Bài 1:** Đọc vào n số nguyên từ bàn phím.

a. Đếm các số chia hết cho 3

b. Đếm các số lớn hơn 10



▪ **Bài 2:** Cho dãy n số nguyên A1, A2... An

a. Tìm các số bằng số trước nó cộng 3

b. Tìm các số bộ ba số trong dãy thỏa mãn điều kiện

$$A_i = A_{i-1} + A_{i+1}$$



TÌM CHỖ CHƯA ĐÚNG!



```
float a[50];  
a[100] = 12.34;
```



```
int a[5] = (0; 1; 2; 3; 4; 5);  
char kt[6] = {a, b, c, d, e, f};
```



```
int arr[10];  
for(int i=1; i<=10; i++)  
    cout<<arr[i];
```





ĐOẠN CHƯƠNG TRÌNH SAU HIỂN THỊ GÌ TRÊN MÀN HÌNH!



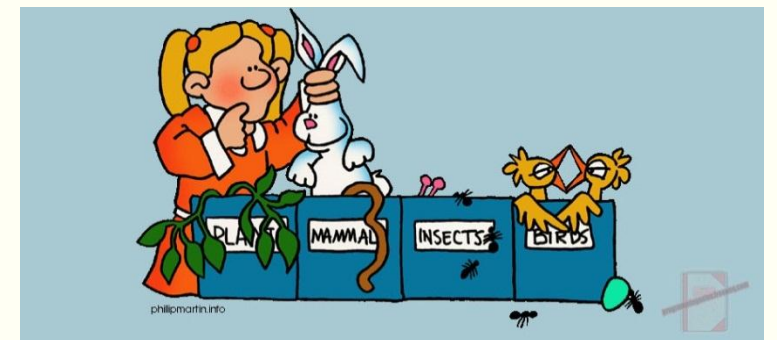
```
char a[3] ={'a', 'b', 'c'};  
for(int i=0; i<3; i++)  
{  
    cout<<a[i]<<" ";  
}
```



```
double a[3] ={1.1, 2.2, 3.3};  
cout<<a[0]<<" "<<a[1] <<" "<<a[2]<<endl;  
a[1]= a[2];  
cout<<a[0]<<" "<<a[1] <<" "<<a[2]<<endl;
```

MỘT SỐ BÀI TOÁN

- Tìm giá trị lớn nhất, nhỏ nhất trong dãy số.
- Sắp xếp dãy số tăng dần (giảm dần)
- Chèn thêm một số vào dãy số ban đầu
- Xóa một phần tử của dãy số



TÌM GIÁ TRỊ LỚN NHẤT

- Bài toán:

- Nhập vào dãy n số nguyên ($n \leq 100$)
- Đưa ra màn hình giá trị lớn nhất trong dãy vừa nhập

TÌM GIÁ TRỊ LỚN NHẤT

```
#include <iostream>
using namespace std;

int main ()
{
    int a[100], n;
    cout<<"Nhap so phan tu cua day (<100) ="; cin>>n;
    for(int i=0; i<n; i++)
    {
        cout<<"a["<<i<<"="; cin>>a[i];
    }

    int max = a[0];
    for(int i=1; i<n; i++)
        if(max< a[i])
            max = a[i];
    cout<<"Gia tri lon nhat cua day la: "<<max;
    return 0;
}
```


SẮP XẾP MẢNG

▪ Bài toán:

Lập trình nhập vào một mảng gồm n phần tử. Sắp xếp các phần tử của mảng theo thứ tự tăng dần. Hiển thị mảng đã sắp xếp ra màn hình.

12	9	24	35	11	18
----	---	----	----	----	----



9	11	12	18	24	35
---	----	----	----	----	----

SẮP XẾP MẢNG

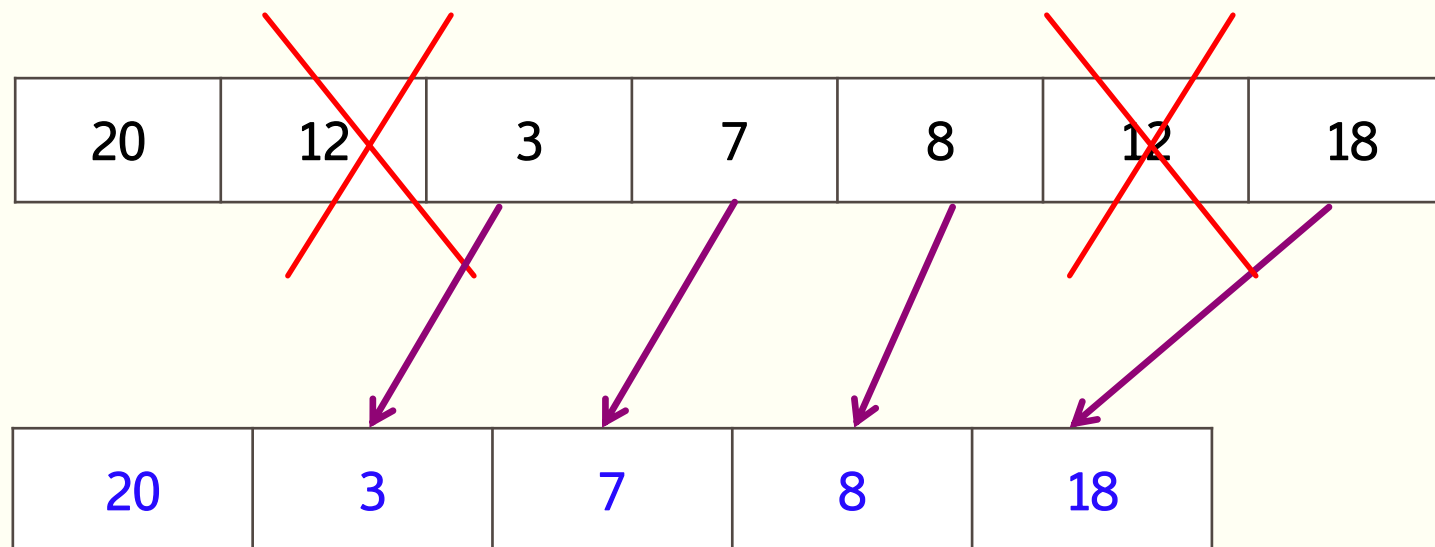
```
#include <iostream>
using namespace std;
int main ()
{
    int a[100], n, temp, i;
    cout<<"Nhap so phan tu cua day (<100) ="; cin>>n;
    for(i=0; i<n; i++)
    {
        cout<<"a["<<i<<"="; cin>>a[i];
    }
    int min, k;
    for(i=0; i<n-1; i++)
        for(int j=i+1; j<n; j++)
            if(a[j] < a[i])
            {
                temp = a[i];
                a[i] = a[j];
                a[j] = temp;
            }
}
```

```
cout<<"Sau khi sap xep:"<<endl;
for(i=0; i<n; i++)
    cout<<a[i]<<" ";
return 0;
}
```

XÓA PHẦN TỬ KHỎI MẢNG

▪ Bài toán:

Lập trình nhập vào một mảng gồm n phần tử. Nhập vào một giá trị cần xóa khỏi mảng. Hiển thị mảng sau khi đã xóa.



XÓA PHẦN TỬ KHỎI MẢNG

```
#include <iostream>
using namespace std;
int main ()
{
    int a[100], n, m, i, x;
    cout<<"Nhap so phan tu cua day (<100) ="; cin>>n;
    for(i=0; i<n; i++)
    {
        cout<<"a["<<i<<"="; cin>>a[i];
    }
    cout<<"Gia tri can xoa:"; cin>>x;
    i=0;m=n;
    while(i<m)
        if(a[i]==x)
        {
            for(int j=i; j<m-1; j++)
                a[j] = a[j+1];
            m--;
        }
        else
            i++;
}
```

```
if(m==n)
    cout<<"Khong co gia tri "<<x<<"
    trong day.";
else
{
    cout<<"Day moi la: "<<endl;
    for(i=0; i<m; i++)
        cout<<a[i]<<" ";
}
return 0;
}
```

MẢNG MỘT CHIỀU

▪ Truyền mảng vào hàm:

Định nghĩa hàm

Truyền mảng vào hàm

```
#include <iostream>
using namespace std;
void nhap(int a[],int n)
{
    for(int i=0; i<n; i++)
    {
        cout<<"a["<<i<<"="; cin>>a[i];
    }
}
void hienThi(int a[],int n)
{
    for(int i=0; i<n; i++)
        cout<<"a["<<i<<"="<<a[i]<<endl;
}
int main ()
{
    int a[100], n;
    cout<<"Nhap so phan tu cua day (<100) ="; cin>>n;
    nhap(a, n) ;
    cout<<"Day so vua nhap:"<<endl;
    hienThi(a, n);
    return 0;
}
```



BÀI TẬP

- **Bài 3:** Đọc vào n số nguyên từ bàn phím. Xóa bỏ số ở vị trí thứ k của dãy, với k đọc vào từ bàn phím. Đưa dãy đã xóa ra màn hình.

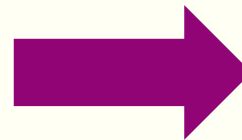


- **Bài 4:** Đọc vào n số ($1 < n < 50$). Đọc thêm một số x , chèn số x vào vị trí thứ 2 của dãy. Đưa dãy sau khi chèn ra màn hình

MẢNG HAI CHIỀU

- Mảng dữ liệu có thể có nhiều chiều:

$$\begin{bmatrix} 3 & 4 & 5 \\ 2 & 7 & 6 \\ 3 & 4 & 9 \end{bmatrix}$$



MA TRẬN



	0	1	2
0	3	4	5
1	2	7	6
2	3	4	9



$$a_{12} = 6$$

MẢNG HAI CHIỀU

▪ Khai báo:

```
kieudulieu tenmang [sohang][socot];
```

▪ Ví dụ:

```
int x[3][5];
```

```
float a[10][20];
```

▪ Khởi tạo giá trị:

```
int x[3][4] = {1, 3, 5, 7, 2, 4, 6, 8, 10, 20, 30, 40}
```

```
int x[3][2] = { {1, 2} , {3, 4} , {5, 6} }
```

```
int a[][] = { {2, 4, 5}, {3, 5, 9} , {7, 2, 8} }
```


MẢNG HAI CHIỀU

▪ Ví dụ:

```
#include <iostream>
using namespace std;
int main ()
{
    int a[20][30]; //khai bao ma tran
    int i, j, cot, hang;
    cout<<"Nhap so hang, cot cua ma tran: ";
    cin>>hang>>cot;
    for(i=0; i< hang; i++)//nhap gia tri cho mang
        for(j=0; j<cot; j++)
        {
            cout<<"a["<<i<<"]["<<j<<"]="; cin>>a[i][j];
        }
    cout<<"\nMa tran vua nhap la:"<<endl;
    for(i=0; i< hang; i++)//hien thi
    {
        for(j=0; j<cot; j++)
            cout<<a[i][j]<<" ";
        cout<<endl;
    }
}
```

MẢNG HAI CHIỀU

- Cộng 2 ma trận cùng kích cỡ:

$$\begin{bmatrix} 3 & 4 & 5 \\ 2 & 7 & 6 \\ 3 & 4 & 9 \end{bmatrix}$$

+

$$\begin{bmatrix} 2 & 1 & 3 \\ 4 & 7 & 9 \\ 5 & 2 & 8 \end{bmatrix}$$



$$\begin{bmatrix} 5 & 5 & 8 \\ 6 & 14 & 15 \\ 8 & 6 & 17 \end{bmatrix}$$

MẢNG HAI CHIỀU

```
#include <iostream>
using namespace std;

int main ()
{
    int a[30][30], b[30][30], c[30][30];
    int i, j, cot, hang;
    cout<<"Nhap so hang, cot cua ma tran: ";
    cin>>hang>>cot;
    cout<<"Nhap ma tran A"<<endl;
    for(i=0; i< hang; i++)//nhap gia tri cho mang A
        for(j=0; j<cot; j++)
        {
            cout<<"a["<<i<<"]["<<j<<"]=""; cin>>a[i][j];
        }
    cout<<"Nhap ma tran B"<<endl;
    for(i=0; i< hang; i++)//nhap gia tri cho mang B
        for(j=0; j<cot; j++)
        {
            cout<<"b["<<i<<"]["<<j<<"]=""; cin>>b[i][j];
        }
}
```

```
cout<<"\nMa tran tong la:"<<endl;
for(i=0; i< hang; i++)//ma tran tong
{
    for(j=0; j<cot; j++)
    {
        c[i][j]= a[i][j] + b[i][j];
        cout<<c[i][j]<<" ";
    }
    cout<<endl;
}
}
```



BÀI TẬP

- **Bài 5:** Lập chương trình nhập từ bàn phím ma trận kích thước m, n ($m \leq 20$, $n \leq 30$). Tính ma trận S cùng kích thước, trong đó:

$$S_{i,j} = \begin{cases} 1 & \text{nếu } a_{i,j} > 0 \\ 0 & \text{nếu } a_{i,j} = 0 \\ -1 & \text{nếu } a_{i,j} < 0 \end{cases}$$

Đưa ra màn hình ma trận S .



- **Bài 6:** Đọc vào ma trận A có m dòng, n cột, các phần tử là số nguyên. Tính tổng các phần tử ở dòng 1. Đọc vào số nguyên k , $k < n$, tính tổng các phần tử ở cột k . Đưa kết quả ra màn hình.



CHUẨN BỊ CHO BÀI SAU!

- Làm việc với vector và struct