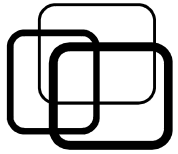
# Mảng nhiều chiều
# Kiểu cấu trúc

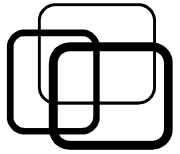
GV. Nguyễn Minh Huy

# Nội dung



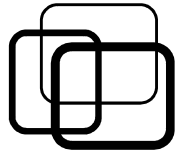
- Mảng nhiều chiều.
- Kiểu cấu trúc.

# Nội dung



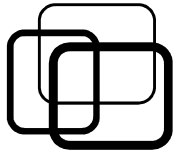
- **Mảng nhiều chiều.**
- Kiểu cấu trúc.

# Mảng nhiều chiều



- Xét chương trình sau:
  - Nhập và xuất ma trận 5 x 10.
    - Khai báo 5 mảng: int a1[10], a2[10], a3[10], a4[10], a5[10].
  - Nhập và xuất ma trận 50 x 10.
    - Khai báo 50 mảng!!
  - ➔ Làm sao biểu diễn ma trận M x N?

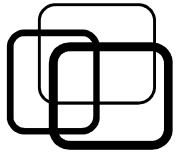
# Mảng nhiều chiều



- Giải pháp 1:
  - Dùng mảng một chiều!!
  - Biểu diễn ma trận M x N:
    - Khai báo mảng một chiều M x N phần tử.
    - Để truy xuất dòng i cột j

      ➔ Truy xuất phần tử [ i * N + j ].

# Mảng nhiều chiều

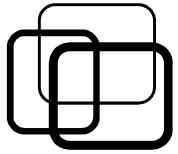


- Giải pháp 2:
  - Dùng mảng của mảng.
  - Từ khóa **typedef**:
    - Đặt tên khác cho kiểu dữ liệu.
    - Cú pháp:

**typedef** <Kiểu dữ liệu> **<Tên khác>**;

```
typedef  int    SoNguyen;
typedef  float  MangThuc[ 10 ];

SoNguyen  a, b, c;
MangThuc  m;

printf("%d %d %d", a, b, c);
m[ 5 ] = 123;
```

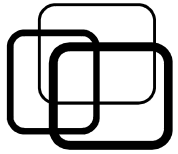
# Mảng nhiều chiều



- Giải pháp 2:
  - Biểu diễn ma trận M x N:
    - Đặt tên cho mảng một chiều N phần tử.
    - Khai báo mảng một chiều M phần tử kiểu đã đặt tên.
    - Để truy xuất dòng i cột j

      ➔ Truy xuất phần tử [ i ] [ j ].

# Mảng nhiều chiều



- Giải pháp 3:
  - Dùng mảng nhiều chiều.
  - Khai báo:

    <Kiểu dữ liệu> **<Tên mảng>[**<Số dòng>**] [**<Số cột>**]**;

    <Số dòng>, <Số cột> phải là một hằng số.

    ```
    int     m1[ 5 ][ 10 ];  // Ma trận 5 x 10 số nguyên.
    int     m2[ M ][ N ];   // Sai.
    ```

  - Truy xuất phần tử:

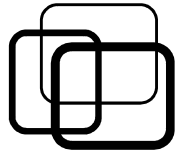
    <Tên mảng> **[ <Chỉ số dòng> ] [ <Chỉ số cột> ]**

    <Chỉ số dòng>: một số nguyên từ **0** đến **<Số dòng> - 1**.

    <Chỉ số cột>: một số nguyên từ **0** đến **<Số cột> - 1**.

    ```
    m1[ 0 ][ 2 ] = 5;
    m1[ 1 ][ 3 ] = 6;
    m1[ -1 ][ 10 ] = 7;    // Sai.
    ```

# Mảng nhiều chiều



- Giải pháp 3:
  - Dùng mảng nhiều chiều.
  - Khởi tạo:

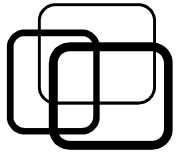
<Kiểu dữ liệu> **<Tên mảng>[**<Số dòng>**] [**<Số cột>**]** =
{
<Khởi tạo dòng 0>,
<Khởi tạo dòng 1>,
…
};

```
// Khởi tạo tất cả phần tử.
int  m1[ 3 ][ 5 ] =
{
        { 1, 1, 1, 1, 1 },
        { 1, 2, 3, 4, 5 },
        { 5, 4, 3, 2, 1 }
};
```

```
// Khởi tạo vài phần tử.
int  m1[ 3][ 5 ] =
{
        { 1, 1 },
        { 1, 2, 3 },
        { 0 }
};
```

```
// Tự động biết số dòng.
int  m1[  ][ 5 ] =
{
        { 1, 1 },
        { 1, 2, 3 },
        { 0 }
};
```

# Khái niệm mảng



- Giải pháp 3:
  - Dùng mảng nhiều chiều.
  - Truyền tham số mảng:
    - Khai báo tham số mảng giống biến mảng.

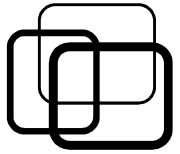
      void foo( **int  a[5][10]** );
    - Khai báo tham số mảng có thể bỏ số dòng.

      void foo( **int  a[ ][10]** );
    - Phần tử mảng có thể bị thay đổi sau khi ra khỏi hàm.

```
void foo( int a[ ][10] )
{
        a[2][2] = 9;
        a[2][5] = 8;
}
```

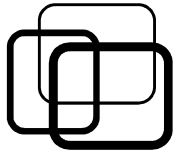
```
void main()
{
        int  a[5][10] = { { 0 } };

        foo(a);
        // a[2][2], a[2][5] bị thay đổi.
}
```

# Nội dung



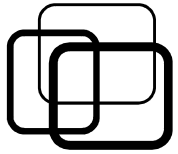
- Mảng nhiều chiều.
- **Kiểu cấu trúc.**

# Kiểu cấu trúc



- Xét chương trình sau:
  - Thông tin một học sinh gồm:
    - Họ tên.
    - Ngày sinh.
    - Giới tính.
    - Điểm văn, toán, ngoại ngữ.
  - Viết chương trình:
    - Nhập vào 1 học sinh.
    - Xuất thông tin học sinh vừa nhập.

# Kiểu cấu trúc



- Kiểu cấu trúc trong C:
  - Kiểu dữ liệu phức hợp.

    ➔ Gom nhóm dữ liệu với nhau.
  - Khai báo kiểu cấu trúc:

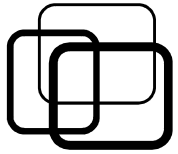
```
struct  <Tên cấu trúc>
{
   <Khai báo thành phần 1>;
   <Khai báo thành phần 2>;
   …
};
```

  - Khai báo biến cấu trúc:

```
<Tên cấu trúc>  <Tên biến>;
```

```
struct HocSinh
{
        char  hoten[50];
        char  ngaysinh[11];
        bool  gioitinh;
        float  diemvan;
        float  diemtoan;
};

void main()
{
        HocSinh  hs1, hs2;
}
```

# Kiểu cấu trúc



- Kiểu cấu trúc trong C:
  - Khởi tạo biến cấu trúc:

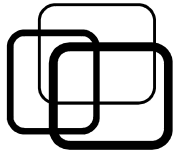
**<Tên cấu trúc>** <Tên biến> =
{
  <Giá trị thành phần 1>,
  <Giá trị thành phần 2>,
  …
};

  - Truy xuất thành phần:

<Tên biến> **. <Tên thành phần>**.

```
void main()
{
        HocSinh  hs =
        {
          “Nguyen Van A”,
          “01/01/1997”,
          1,
          7, 8, 9
        };

        hs.diemvan = 5;
        hs.diemToan = 9;
}
```

# Kiểu cấu trúc



- Kiểu cấu trúc trong C:
  - Truyền tham số:
    - Giống các kiểu bình thường.
    - Truyền tham trị

      ➔ Giá trị thành phần không đổi.
    - Truyền tham chiếu

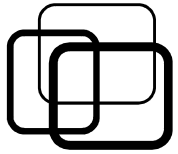
      ➔ Giá trị thành phần thay đổi.

```
void  cong1(HocSinh hs)
{
        hs.diemvan++;
        hs.diemtoan++;
}

 void  cong2(HocSinh &hs)
{
        hs.diemvan++;
        hs.diemtoan++;
}

void main()
{
        HocSinh  hs1;
        cong1(hs1);
        cong2(hs1);
}
```

# Tóm tắt

- Mảng nhiều chiều:
  - Giải pháp 1: mảng một chiều.
  - Giải pháp 2: mảng của mảng.
  - Giải pháp 3: mảng nhiều chiều thật sự.
- Kiểu cấu trúc:
  - Kiểu dữ liệu phức hợp.
  - Gom nhóm dữ liệu với nhau.
  - Khai báo: từ khóa **“struct”**.
  - Truy xuất phần tử: dấu **“.”**.

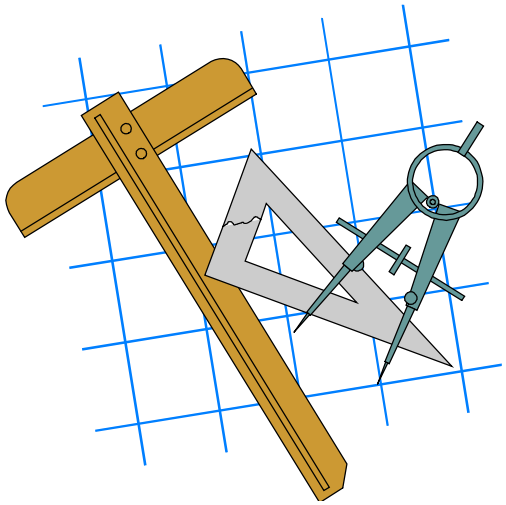
# Bài tập



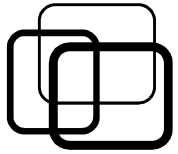
- **Bài tập 8.1:**

  Viết chương trình C thao tác ma trận như sau:

  - Nhập vào ma trận vuông N x N chứa số nguyên.
  - Hãy cho biết:
    a) Tổng phần tử nằm trên đường chéo chính/phụ.
    b) Dòng có tổng lớn nhất.
    c) Ma trận có là một ma phương hay không.



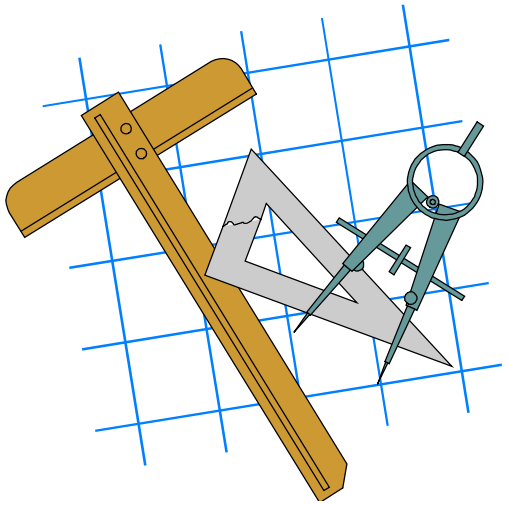
# Bài tập



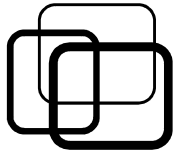
- **Bài tập 8.2:**

  Viết chương trình C thao tác ma trận như sau:

  - Nhập vào ma trận M x N chứa số nguyên.
  - Xuất ra màn hình các phần tử có giá trị bằng tổng các phần tử còn lại trên dòng và cột của nó.



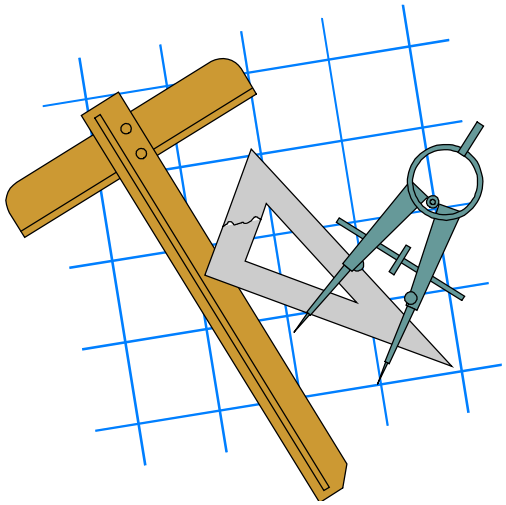
# Bài tập



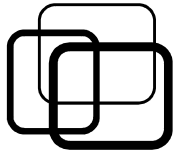
- **Bài tập 8.3:**

  Viết chương trình C thao tác phân số như sau:

  - Khai báo kiểu cấu trúc phân số.
  - Nhập vào 2 phân số.
  - Tính và xuất kết quả tổng, tích, nghịch đảo, rút gọn của 2 phân số vừa nhập.



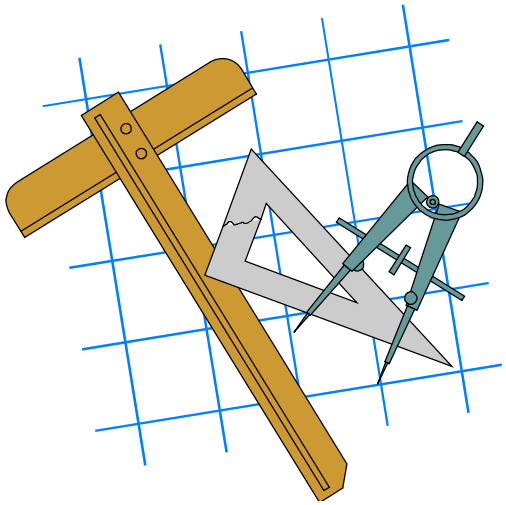
# Bài tập



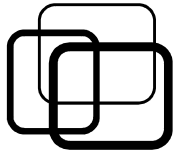
- **Bài tập 8.4:**

  Viết chương trình C thao tác đơn thức như sau:

  - Khai báo kiểu cấu trúc đơn thức (có dạng $ax^N$).
  - Nhập vào 2 đơn thức.
  - Tính và xuất kết quả tích, thương, đạo hàm của hai đơn thức vừa nhập.



# Bài tập



- **Bài tập 8.5:**

  Viết chương trình C thao tác học sinh như sau:

  - Khai báo kiểu cấu trúc học sinh (như bài học).
  - Nhập vào danh sách N học sinh.
  - Xuất danh sách học sinh giỏi (điểm trung bình >= 8) theo thứ tự điểm trung bình giảm dần.