

# Một số vấn đề nâng cao

GV. Nguyễn Minh Huy



- Mạng động.
- Độ quy.
- Các thuật toán thông dụng.



- **Mảng động.**
- **Đệ quy.**
- **Các thuật toán thông dụng.**



## ■ Vấn đề với mảng tĩnh:

- Phải khai báo kích thước tối đa.
- Không phải lúc nào cũng dùng hết.
  - Tốn nhiều bộ nhớ.
  - Nhu cầu mảng kích thước tùy biến.

## ■ Khái niệm con trỏ:

- Biến “lưu động”.
  - Có thể “định cư” ở nhiều vùng nhớ.
  - “Trở” đến các vùng nhớ khác nhau.

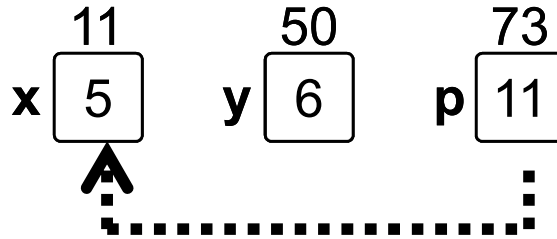


## ■ Con trỏ trong C:

- Khai báo: **<Kiểu dữ liệu> \*<Tên con trỏ>;**

- Lấy địa chỉ biến: **<Tên con trỏ> = &<Tên biến>;**

```
int x = 5;  
int y = 6;  
int *p;  
p = &x;
```



- Truy xuất dữ liệu ô nhớ:

**<Tên biến> = \*<Tên con trỏ>;**

**<Tên biến> = <Tên con trỏ>[<Chỉ số ô nhớ>;**

```
int z = *p;  
int t = p[2];
```



## ■ Con trỏ trong C:

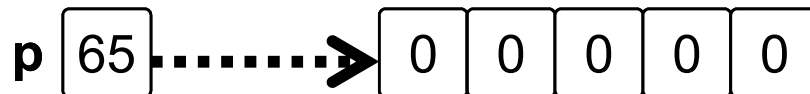
### ■ Lệnh xin cấp vùng nhớ:

- Cú pháp: **calloc**(<số ô nhớ>, <kích thước mỗi ô nhớ>);
- Trả về: địa chỉ vùng nhớ (thành công), NULL (thất bại).
- Địa chỉ trả về phải cùng kiểu với con trỏ nhận.

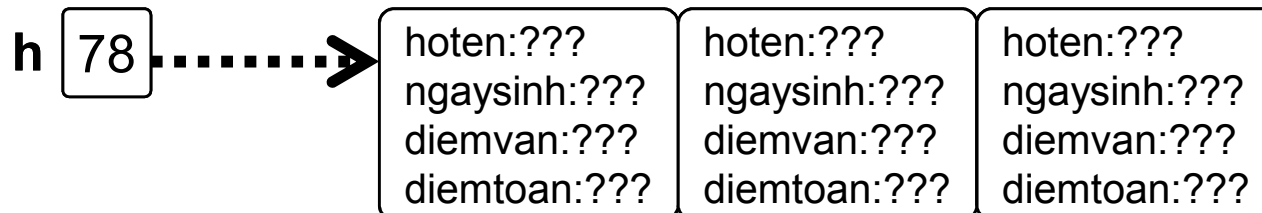
### ■ Lệnh thu hồi vùng nhớ:

- Cú pháp: **free**(<Tên con trỏ>);

```
int *p = (int *) calloc( 5, sizeof( int ) );
```



```
HocSinh *h = (HocSinh *) calloc( 3, sizeof( HocSinh ) );
```



```
free(p);  
free(h);
```



- Mạng động.
- **Đệ quy.**
- Các thuật toán thông dụng.



## ■ Khái niệm đệ quy:

- Định nghĩa một khái niệm dựa trên chính nó.
- Cấu trúc:
  - Phần định nghĩa trường hợp cơ bản (trường minh).
  - Phần định nghĩa đệ quy.
- Ví dụ: định nghĩa lũy thừa  $x^n$ 
  - $LT(x, 0) = 1$ .
  - $LT(x, n) = x * LT(x, n - 1), \quad n > 0$ .
- Ưu điểm:
  - Nêu bản chất vấn đề.
  - Ngắn gọn, dễ hiểu.





## ■ Hàm đệ quy:

- Dùng cài đặt khái niệm đệ quy.

- Cấu trúc:

```
<Kiểu trả về> <Tên hàm>(<Danh sách tham số>)  
{  
    if (<Trường hợp cơ bản>)  
        <Xử lý trường hợp cơ bản>;  
    else  
        <Gọi lại hàm đệ quy>;  
}
```

```
long luythua(float x, int n)  
{  
    if ( n == 0 )  
        return 1;  
    return x * luythua(x, n - 1);  
}
```



- Mạng động.
- Độ quy.
- **Các thuật toán thông dụng.**



## ■ Bài toán sắp xếp:

- Nhập vào mảng N số nguyên.
- Sắp xếp và xuất mảng theo thứ tự tăng dần.
  - ➔ Thuật toán Interchange Sort.
  - ➔ Thuật toán Selection Sort.
  - ➔ Thuật toán Quicksort.
- ...



## ■ Bài toán sắp xếp:

### ■ Nghịch thế:

- Cặp phần tử đứng không đúng thứ tự.
- $(a_i, a_j)$  nghịch thế  $\Leftrightarrow i < j$  và  $a_i > a_j$ .

### ■ Thuật toán Interchange Sort:

- Xét tất cả các cặp phần tử mảng.
- Với mỗi cặp, hoán vị chúng nếu nghịch thế.



## ■ Bài toán tìm kiếm:

### ■ Trò chơi đoán số:

- B1: người chơi nghĩ ra một số nguyên  $N$  ( $1 \leq N \leq 100$ ).
- B2: máy đưa ra dự đoán một số nguyên  $K$ .
- B3: người chơi cho biết  $K$  lớn hơn, nhỏ hơn hay bằng  $N$ .
- B4: nếu bằng → kết thúc trò chơi.  
ngược lại → quay lại B2.

Hãy đề ra một thuật toán để số lần đoán của máy là ít nhất.

→ Tìm kiếm tuần tự:  $O(n)$ .

→ Tìm kiếm nhị phân:  $O(\log_2 n)$ .



## ■ Bài toán tìm kiếm:

### ■ Tìm kiếm nhị phân:

- B1: đặt (left, right) là đoạn cần tìm.
- B2: **nếu** chiều dài đoạn cần tìm = 0 → kết thúc thất bại.

**ngược lại**

B2.1: kiểm tra phần tử giữa đoạn.

B2.2: **nếu** thỏa điều kiện tìm → kết thúc thành công.

**ngược lại**

B2.2.1: thu hẹp một nửa đoạn cần tìm  
(dựa vào phần tử giữa).

B2.2.2: quay lại B2.



## ■ Con trỏ:

- Biến “tham chiếu”, “trỏ” đến các ô nhớ.
- Toán tử &: lấy địa chỉ ô nhớ.
- Toán tử \*: truy xuất ô nhớ.

## ■ Độ quy:

- Định nghĩa một khái niệm dựa trên chính nó.
- Hàm đệ quy: hàm gọi lại chính mình.

## ■ Thuật toán thông dụng:

- Sắp xếp Interchange Sort.
- Tìm kiếm nhị phân.





## ■ Bài tập 10.1:

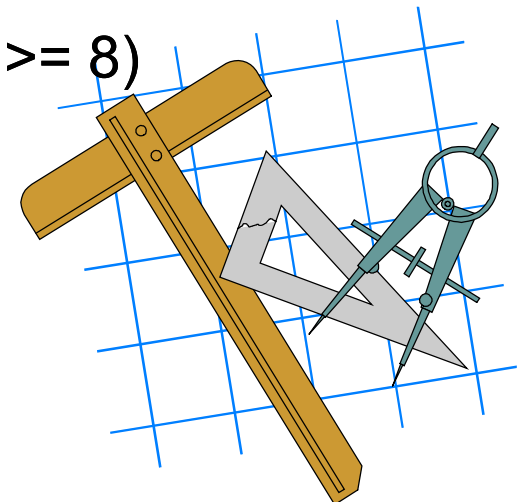
Thông tin một học sinh bao gồm: họ tên, điểm văn, điểm toán.

File input.txt chứa danh sách học sinh như sau:

- Dòng đầu tiên chứa: N (số học sinh trong danh sách).
- N dòng tiếp theo, dòng i ( $1 \leq i \leq N$ ) chứa:  $hoten_i$   $van_i$   $toan_i$  (thông tin học sinh thứ i trong danh sách).

Viết chương trình C (sử dụng kiểu cấu trúc và mảng động):

- Nhập danh sách học sinh từ file input.txt
- Xuất danh sách học sinh giỏi (điểm trung bình  $\geq 8$ ) theo thứ tự điểm trung bình giảm dần vào file output.txt (cấu trúc tương tự file input.txt).



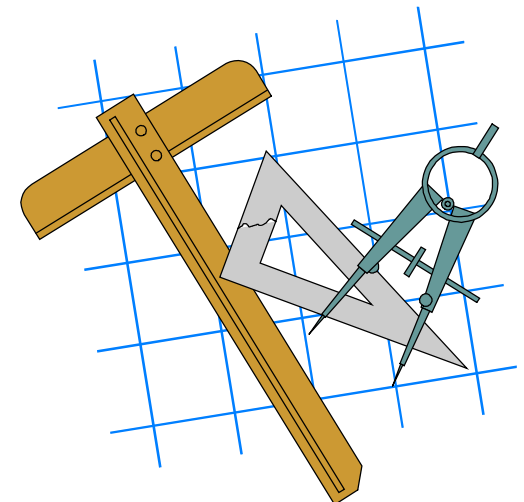




## ■ Bài tập 10.2:

Viết chương trình C tìm ước chung lớn nhất (sử dụng đệ quy) như sau:

- Nhập vào hai số nguyên dương  $a$ ,  $b$ .
- Tìm và xuất ước chung lớn nhất của  $a$ ,  $b$  (thuật toán Euclid).





## ■ Bài tập 10.3:

Viết chương trình C tính dãy Fibonacci (sử dụng đệ quy) như sau:

- Nhập vào số nguyên  $N \geq 0$ .
- Xuất các số Fibonacci từ  $F_0$  đến  $F_N$ .

