

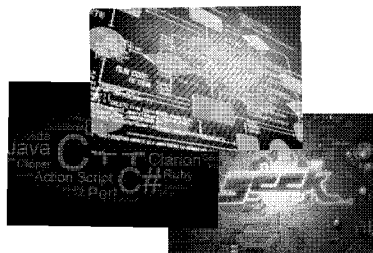


ĐẠI HỌC QUỐC GIA TP HCM
TRƯỜNG ĐẠI HỌC
CÔNG NGHỆ THÔNG TIN

NHẬP MÔN LẬP TRÌNH

CHƯƠNG VI

MẢNG VÀ CẤU TRÚC



Nguyễn Trọng Chính
chinhnt@uit.edu.vn

MẢNG VÀ CẤU TRÚC

- ❖ MẢNG CỐ ĐỊNH
- ❖ MỘT SỐ VẤN ĐỀ LẬP TRÌNH VỚI MẢNG
- ❖ CHUỖI KÝ TỰ
- ❖ DỮ LIỆU CÓ CẤU TRÚC

MỘT SỐ VẤN ĐỀ LẬP TRÌNH VỚI MẢNG

❖ SẮP XẾP TRÊN MẢNG

Sắp xếp là việc hoán đổi vị trí của các phần tử thứ i và j trong mảng A có n phần tử sao cho quan hệ về vị trí và giá trị của chúng thỏa một quan hệ thứ tự R nào đó. Khi đó, mảng sẽ có thứ tự là R .

$$A[i] R A[j] \quad \forall i < j \quad (i, j = 0..n-1)$$

Ví dụ: Mảng có thứ tăng dần thì R là quan hệ \leq và các phần tử trong mảng có các quan hệ như sau:

$$A[i] \leq A[j] \quad \forall i < j \quad (i, j = 0..n-1)$$

3

MỘT SỐ VẤN ĐỀ LẬP TRÌNH VỚI MẢNG

❖ SẮP XẾP TRÊN MẢNG

Giả sử mảng A chứa n phần tử có thứ tự R . Khi đó, $A[i]$ và $A[j]$ là một nghịch thế nếu

$$A[j] R A[i], \quad i < j \quad (i, j = 0..n-1)$$

Ví dụ: Mảng có A thứ tăng dần thì R là quan hệ \leq và $A[i], A[j]$ là một nghịch thế nếu:

$$A[j] \leq A[i] \quad \forall i < j \quad (i, j = 0..n-1)$$

4

MỘT SỐ VẤN ĐỀ LẬP TRÌNH VỚI MẢNG

❖ SẮP XẾP TRÊN MẢNG

Nguyên tắc sắp xếp mảng là hoán vị các phần tử nhằm triệt tiêu tất cả các nghịch thế trong mảng.

* **Phương pháp nổi bọt (bubble sort)**: Duyệt các phần tử $A[i]$, $i = 0..n-2$, nếu $A[i+1] > A[i]$ thì đổi chỗ phần tử $A[i]$ với phần tử $A[i+1]$. Thực hiện việc duyệt đến khi A không chứa nghịch thế.

5

MỘT SỐ VẤN ĐỀ LẬP TRÌNH VỚI MẢNG

❖ SẮP XẾP TRÊN MẢNG

Ví dụ: sắp xếp mảng có giá trị tăng dần

```
const int MAX = 100;
int a[MAX], n, m, i, t;
//...
do {
    m = 0;
    for (i = 0; i < n - 1; i++)
        if (a[i] > a[i + 1]) {t = a[i]; a[i] = a[i+1]; a[i+1] = t; m++;}
} while (m);
```

6

MỘT SỐ VẤN ĐỀ LẬP TRÌNH VỚI MẢNG

❖ SẮP XẾP TRÊN MẢNG

* Phương pháp đổi chỗ trực tiếp (interchange sort): Duyệt các phần tử $A[i]$, $i=0..n-2$, so sánh $A[i]$ với $A[j]$, $j=i+1,..n-1$, nếu $A[j] < A[i]$ thì đổi chỗ $A[i]$ với $A[j]$.

7

MỘT SỐ VẤN ĐỀ LẬP TRÌNH VỚI MẢNG

❖ SẮP XẾP TRÊN MẢNG

Ví dụ: sắp xếp mảng có giá trị tăng dần

```
const int MAX = 100;
int a[MAX], n, i, j, t;
//...
for (i = 0; i < n - 1; i++)
    for (j = i + 1; j < n; j++)
        if (a[i] > a[j]) {
            t = a[i]; a[i] = a[j]; a[j] = t;
        }
```

8

MỘT SỐ VẤN ĐỀ LẬP TRÌNH VỚI MẢNG

❖ SẮP XẾP TRÊN MẢNG

* Phương pháp chọn trực tiếp (selection sort):

Chọn phần tử thứ i của A , $i = 0..n-2$ sao cho

$$A[i] \leq A[j], \forall j=i..n-1$$

9

MỘT SỐ VẤN ĐỀ LẬP TRÌNH VỚI MẢNG

❖ SẮP XẾP TRÊN MẢNG

Ví dụ: sắp xếp mảng có giá trị tăng dần

```
const int MAX = 100;
int a[MAX], n, m, i, j, t;
//...
for (i = 0; i < n - 1; i++) {
    m = i;
    for (j = i + 1; j < n; j++)
        if (a[m] > a[j]) m = j;
    if (m != i) { t = a[m]; a[m] = a[i]; a[i] = t; }
}
```

10

MỘT SỐ VẤN ĐỀ LẬP TRÌNH VỚI MẢNG

❖SẮP XẾP TRÊN MẢNG

* Phương pháp phân đoạn (quick sort):

- Nếu mảng cần sắp xếp có ít hơn 2 phần tử thì mảng đã có thứ tự.
- Chọn một phần tử trong mảng làm phần tử gốc
- Chia mảng thành 2 đoạn, đoạn bên trái chứa các phần tử thỏa quan hệ R với phần tử gốc, đoạn bên phải chứa các phần tử mà phần tử gốc thỏa quan hệ R với chúng.
- Thực hiện việc sắp xếp trên 2 đoạn con.

11

MỘT SỐ VẤN ĐỀ LẬP TRÌNH VỚI MẢNG

❖SẮP XẾP TRÊN MẢNG

Ví dụ: sắp xếp mảng có giá trị tăng dần

```
void quicksort(int a[], int left, int right) {  
    int mi = left, i, t;  
    if (left >= right) return;  
    for (i = left; i < right; i++)  
        if (a[i] < a[right]) { t = a[mi]; a[mi] = a[i]; a[i] = t; mi++; }  
    t = a[mi]; a[mi] = a[right]; a[right] = t;  
    quicksort(a, left, mi - 1);  
    quicksort(a, mi + 1, right);  
}
```

12

MỘT SỐ VẤN ĐỀ LẬP TRÌNH VỚI MẢNG

❖THÊM MỘT PHẦN TỬ VÀO MẢNG

* Thêm phần tử a vào cuối mảng:

- Gán giá trị a cho phần tử có chỉ số n
- Tăng n lên 1.

* Thêm phần tử a vào vị trí có chỉ số i:

- Di chuyển các phần tử j ($j=i..n-1$) qua phải 1 vị trí.
- Gán giá trị a cho phần tử có chỉ số i
- Tăng n lên 1.

13

MỘT SỐ VẤN ĐỀ LẬP TRÌNH VỚI MẢNG

❖XÓA PHẦN TỬ CỦA MẢNG

* Xóa phần tử cuối mảng:

- Giảm n đi 1.

* Xóa phần tử ở vị trí có chỉ số i:

- Di chuyển các phần tử j ($j=i+1..n-1$) qua trái 1 vị trí.
- Giảm n đi 1.

14

MỘT SỐ VẤN ĐỀ LẬP TRÌNH VỚI MẢNG

❖ TÌM KIẾM TRÊN MẢNG

* Tìm kiếm tuần tự: Dùng để tìm kiếm trên mảng chưa có thứ tự

- Duyệt tuần tự các phần tử trong mảng.
- Nếu phần tử đang xét là phần tử đang tìm, trả về chỉ số của phần tử đó và kết thúc tìm kiếm.

15

MỘT SỐ VẤN ĐỀ LẬP TRÌNH VỚI MẢNG

❖ TÌM KIẾM TRÊN MẢNG

* Tìm kiếm nhị phân: Dùng để tìm kiếm trên mảng đã có thứ tự R

- Nếu mảng không có phần tử nào thì kết thúc. Kết quả không tìm thấy.
- Nếu phần tử cần tìm X ở vị trí chính giữa mảng, trả về chỉ số của vị trí này và kết thúc.
- Nếu phần tử cần X tìm thỏa quan hệ R với phần tử M ở vị trí chính giữa mảng, tìm X trong phần mảng bên trái của M. Ngược lại, tìm X trong phần mảng bên phải của M.

16

CHUỖI KÝ TỰ

❖CẤU TRÚC CỦA MỘT CHUỖI

- Chuỗi trong C/C++ là một mảng các phần tử có kiểu char.
- Chuỗi kết thúc bằng một ký tự '\0'.
- Độ dài chuỗi được tính từ vị trí bắt đầu của mảng đến vị trí liền trước ký tự '\0'.

17

CHUỖI KÝ TỰ

❖KHAI BÁO CHUỖI

`char tên_chuỗi[N];`

`char tên_chuỗi[N] = "giá trị";`

`char tên_chuỗi[N] = {'g','i','á','t','r','ì','\0'};`

Lưu ý:

- N là số lượng ký tự bao gồm cả ký tự '\0';
- Cần xác định N là số lượng ký tự lớn nhất mà chuỗi có thể đạt được cộng thêm 1.

18

BÀI TẬP

- 1) Viết hàm sinh một chuỗi gồm n ký tự c.
- 2) Viết hàm xác định vị trí xuất hiện đầu tiên của chuỗi fs trong chuỗi s cho trước.
- 3) Viết hàm đếm số lượng từ trong chuỗi.
- 4) Viết hàm đổi tất cả ký tự đầu tiên của từ trong chuỗi thành ký tự in hoa.
- 5) Viết chương trình nhập vào một danh sách tên học sinh và sắp xếp theo thứ tự Alphabet của chữ cái đầu tiên của tên học sinh. Trong đó, đoạn code sắp xếp tên học sinh được tổ chức thành một hàm.