Sorting Algorithms

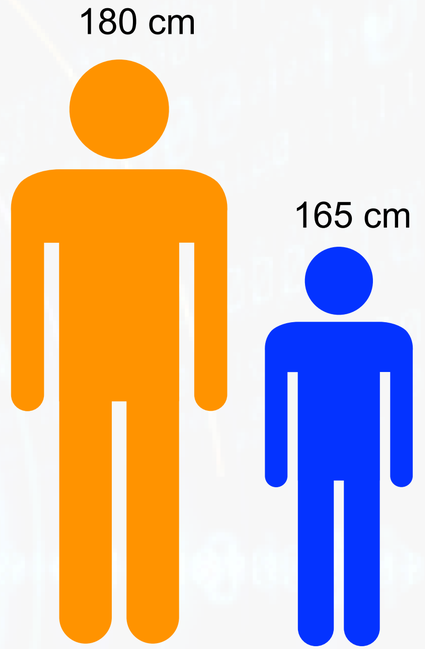
CYBERLEARN
ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH

KHÁI NIỆM- THUẬT TOÁN SẮP XẾP

- Để truy xuất thông tin nhanh chóng và chính xác → thông tin phải được sắp xếp theo một trật tự hợp lý nào đó
- Một số CTDL đã định nghĩa trước trật tự của các phần tử, khi đó mỗi phần tử khi thêm vào phải đảm bảo trật tự này
- Sắp xếp là quá trình xử lý một danh sách các phần tử (hoặc các mẫu tin) để đặt chúng theo một thứ tự thỏa mãn một tiêu chuẩn nào đó dựa trên nội dung thông tin lưu giữ tại mỗi phần tử

CYBERLEARN
ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH

KHÁI NIỆM- THUẬT TOÁN SẮP XẾP



- Khái niệm nghịch thế

a_1	a_2	a_3	a_4	...	...	a_{N-2}	a_{N-1}	a_N

Giả sử xét mảng có thứ tự tăng dần, nếu có $i<j$ và $a_i>a_j$ thì ta gọi đó là nghịch thế.

Mục tiêu của sắp xếp là *khử các nghịch thế* (bằng cách hoán vị)

CYBERLEARN

KHÁI NIỆM- THUẬT TOÁN SẮP XẾP

- Tương tự như các giải thuật tìm kiếm, khối lượng công việc phải thực hiện có liên quan chặt chẽ với số lần so sánh các khóa
- Ngoài ra, các giải thuật sắp xếp còn phải di chuyển các phần tử

→ Chiếm nhiều thời gian khi các phần tử có kích thước lớn

CYBERLEARN
ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH

Đổi chỗ trực tiếp
– Interchange Sort

●Ý tưởng

Xuất phát từ đầu dãy, tìm tất cả nghịch thế chứa phần tử này, triệt tiêu chúng bằng cách đổi chỗ phần tử này với phần tử tương ứng trong cặp nghịch thế. Lặp lại xử lý trên với các phần tử tiếp theo trong dãy.

CYBERLEARN
ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH

Giải thuật

- Bước 1 : i = 0;// bắt đầu từ đầu dãy
- Bước 2 : j = i+1;//tìm các phần tử a[j] < a[i], j>i
- Bước 3 :

  Trong khi j <= N thực hiện

  Nếu a[j]<a[i]: Hoán vị a[i], a[j];

  j = j+1;
- Bước 4 : i = i+1;

  Nếu i < N: Lặp lại Bước 2.

  Ngược lại: Dừng.

CYBERLEARN
ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH

Bài tập

- Minh họa từng bước thực hiện của giải thuật Interchange Sort khi sắp dãy số sau tăng dần:

15	7	9	10	6	20
1	*2*	*3*	*4*	*5*	*6*

- Cho biết tổng số phép hoán vị

Đổi chỗ trực tiếp – Interchange Sort

- **Đánh giá giải thuật**

Số lượng các phép so sánh xảy ra không phụ thuộc vào tình trạng của dãy số ban đầu, nhưng *số lượng phép hoán vị thực hiện tùy thuộc vào kết quả so sánh*, có thể ước lượng:

```
void interchangeSort(int []a )
{
int i, j;
for (i = 0 ; i<a.length-1 ; i++)
{
    for (j =i+1; j < a.length ; j++) {
        if(a[j ]< a[i])
        {
                int tam = a[i];
                a[i] = a[j];
                a[j] = tam;
            }
        }
}
}
```

Trường hợp	Số lần so sánh	Số lần hoán vị
Tốt nhất	$\sum_{i=1}^{n}(n-i+1) = \frac{n(n-1)}{2}$	0
Xấu nhất	$\frac{n(n-1)}{2}$	$\sum_{i=1}^{n-1}(n-i+1) = \frac{n(n-1)}{2}$

CYBERLEARN
ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH

Bubble Sort

SẮP XẾP NỔI BỌT

#SORTING #ALGORITHM

Nổi bọt – Bubble sort

Ý tưởng:

- Xuất phát từ cuối dãy, đổi chỗ các cặp phần tử kế cận để đưa phần tử nhỏ hơn trong cặp phần tử đó về vị trí đúng đầu dãy hiện hành
- Sau đó sẽ không xét đến nó ở bước tiếp theo, do vậy ở lần xử lý thứ i sẽ có vị trí đầu dãy là i
- Lặp lại xử lý trên cho đến khi không còn cặp phần tử nào để xét.

CYBERLEARN
ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH

Giải thuật

Bước 1:

i = 0;

Bước 2:

j = n-1;

Trong khi (j > i) thực hiện:

Nếu a[j]<a[j-1]: Hoán vị a[j] và a[j-1]

j = j-1;

Bước 3:

i = i+1;

Nếu i = n: Dừng

Ngược lại: Lặp lại Bước 2.

CYBERLEARN
ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH

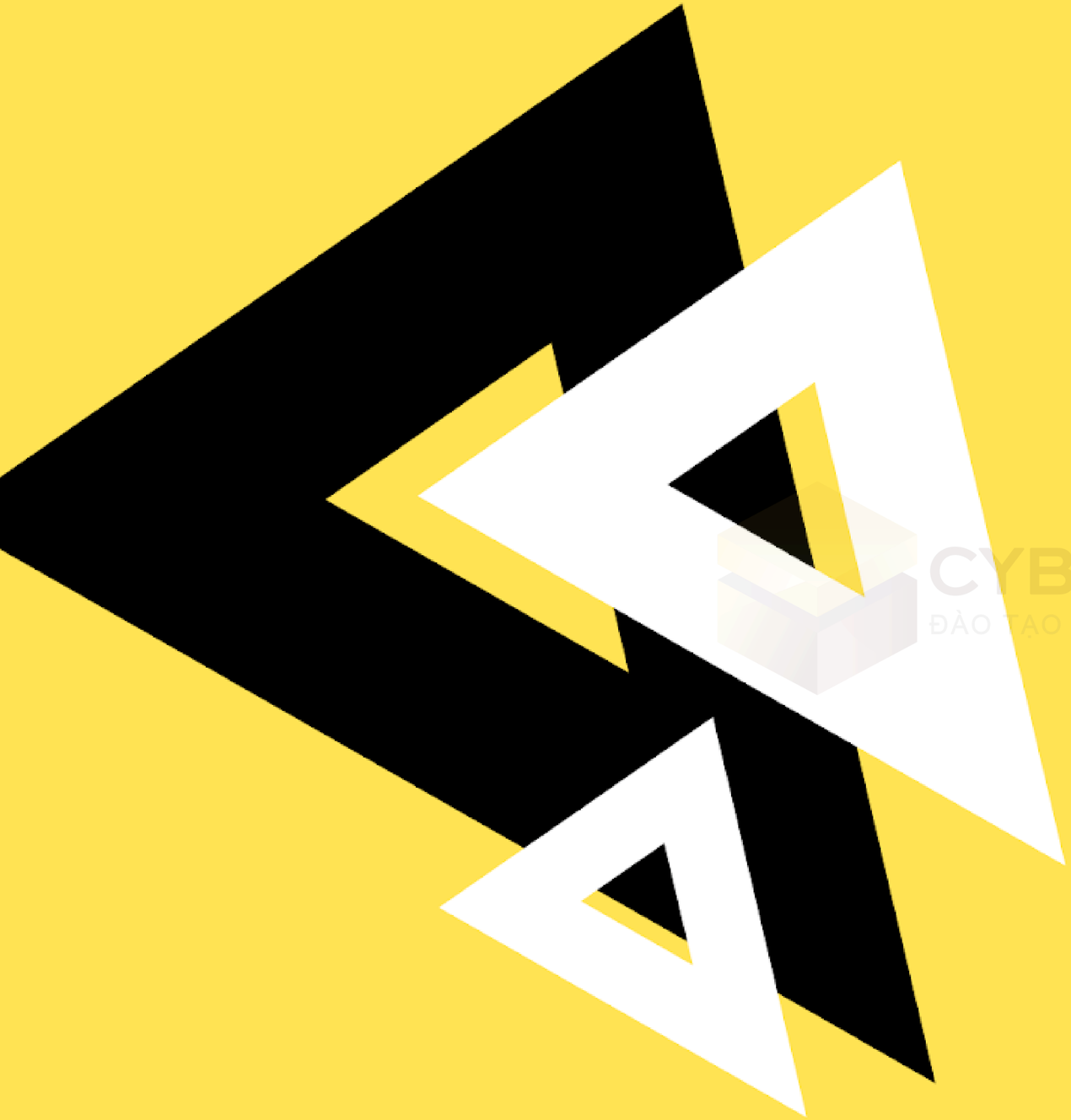
Đánh giá giải thuật

- Trong mọi trường hợp, số phép so sánh là:

$$(n-1) + (n-2) + \ldots + 1 = n(n-1)/2 = \mathbf{O(n^2)}$$

- Số phép hoán vị:
  - Trường hợp xấu nhất: **n(n-1)/2**
  - Trường hợp tốt nhất: **0**

CYBERLEARN
ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH



SELECTION SORT

CHỌN TRỰC TIẾP

CYBERLEARN
ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH

Chọn trực tiếp
Selection sort

Ý tưởng:

- Chọn phần tử nhỏ nhất trong N phần tử ban đầu, đưa phần tử này về vị trí đúng là đầu dãy hiện hành (*vị trí 1*);
- Dãy hiện hành còn lại N-1 phần tử cần sắp xếp (*từ vị trí 2 đến N*), lặp lại quá trình trên cho dãy hiện hành...
- Thực hiện cho đến khi dãy hiện hành chỉ còn 1 phần tử

CYBERLEARN
ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH

CyberSoft.edu.vn - CyberLearn.vn

Chọn trực tiếp
Selection sort

Làm sao để xác định được vị trí phần tử có giá trị nhỏ nhất trong một dãy gồm N phần tử?

CyberSoft.edu.vn - CyberLearn.vn

Giải thuật tìm vị trí phần tử nhỏ nhất

Bước 1:

i=1, vt=0;

Bước 2:

Nếu i>N thì trả về giá trị ***vt****, kết thúc;*

Bước 3:

Nếu a[i]<a[vt] thì vt=i;

i=i+1;

Quay lại Bước 2;

CYBERLEARN

Chọn trực tiếp
Selection sort

Giải thuật

Bước 1: i = 1;

Bước 2: Tìm phần tử a[vtmin] nhỏ nhất trong dãy hiện hành từ a[i] đến a[N]

Bước 3: Hoán vị a[vtmin] và a[i]

Bước 4:

i = i+1

Nếu i < N thì lặp lại Bước 2

Ngược lại: Dừng.

CYBERLEARN
ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH

Bài tập

- Minh họa từng bước thực hiện của giải thuật Selection Sort khi sắp dãy số sau tăng dần:

15	7	9	10	6	20
1	*2*	*3*	*4*	*5*	*6*

- Cho biết tổng số phép gán và so sánh

CYBERLEARN
ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH

Chọn trực tiếp
Selection sort

Đánh giá giải thuật

Đối với giải thuật chọn trực tiếp, có thể thấy rằng ở lượt thứ i, bao giờ cũng cần (n-i) lần so sánh để xác định phần tử nhỏ nhất hiện hành. Số lượng phép so sánh này không phụ thuộc vào tình trạng của dãy số ban đầu, do vậy trong mọi trường hợp có thể kết luận:

Trường hợp	Số lần so sánh	Số phép gán
Tốt nhất	n(n-1)/2	0
Xấu nhất	n(n-1)/2	3n(n-1)/2

CYBERLEARN
ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH

Chèn trực tiếp
- Insertion sort

Ý tưởng

Cho dãy ban đầu $\mathbf{a_1}$, $\mathbf{a_2}$,... ,$\mathbf{a_n}$

a_1	a_2	a_3	a_4	...	a_n

CYBERLEARN
ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH

Đã có đoạn gồm một phần tử $\mathbf{a_1}$ được sắp

a_1	a_2	a_3	a_4	...	a_n

Chèn trực tiếp
- Insertion sort

- Sau đó thêm $\mathbf{a_2}$ vào đoạn $\mathbf{a_1}$ sẽ có đoạn $\mathbf{a_1\ a_2}$ được sắp

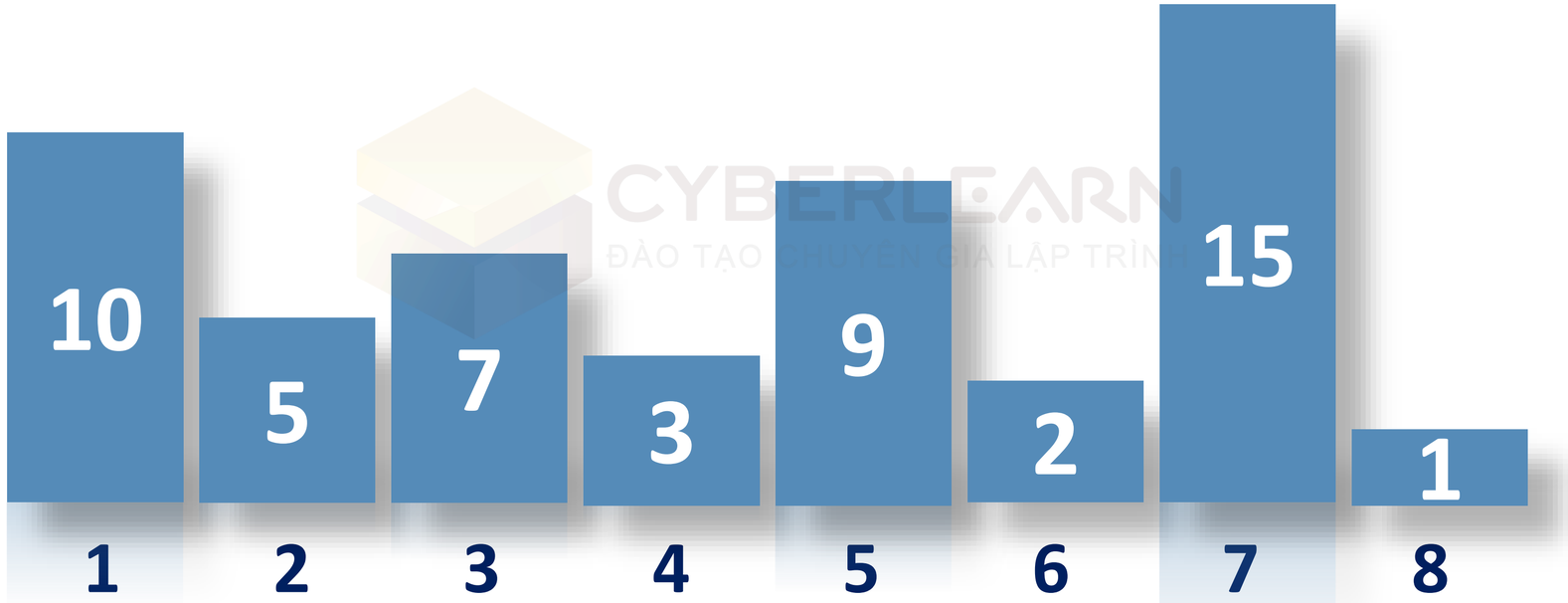
- Tiếp tục thêm $\mathbf{a_3}$ vào đoạn $\mathbf{a_1\ a_2}$ để có đoạn $\mathbf{a_1\ a_2\ a_3}$ được sắp;

- Tiếp tục cho đến khi thêm xong $\mathbf{a_N}$ vào đoạn $\mathbf{a_1\ a_2\ ...a_{N-1}}$ sẽ có dãy $\mathbf{a_1\ a_2\ ....\ a_N}$ được sắp.

CYBERLEARN
ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH
CyberSoft.edu.vn - CyberLearn.vn

Chèn trực tiếp
– Insertion sort

Giả sử cần sắp xếp dãy số sau tăng dần



CYBERLEARN
ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH

Chèn trực tiếp
- Insertion sort

Mô tả giải thuật bằng mã tự nhiên

Lặp

Bước 1: i = 1; // giả sử có đoạn a[0] đã được sắp

Bước 2: x = a[i];

Tìm vị trí *pos* thích hợp trong đoạn *[1..i]* để chèn a[i] vào

Bước 3: Dời chỗ các phần tử từ *pos* đến *i-1* sang phải 1 vị trí để dành chỗ cho a[i]

Bước 4: a[pos] = x; // có đoạn a[0]..a[i] đã được sắp

Bước 5: i = i+1;

Nếu i ≤ N : Lặp lại Bước 2.

Ngược lại : Dừng.

CYBERLEARN
ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH

Bài tập

- Minh họa từng bước thực hiện khi áp dụng giải thuật Insertion Sort để sắp xếp dãy số sau tăng dần

15	7	9	10	6	20
1	*2*	*3*	*4*	*5*	*6*

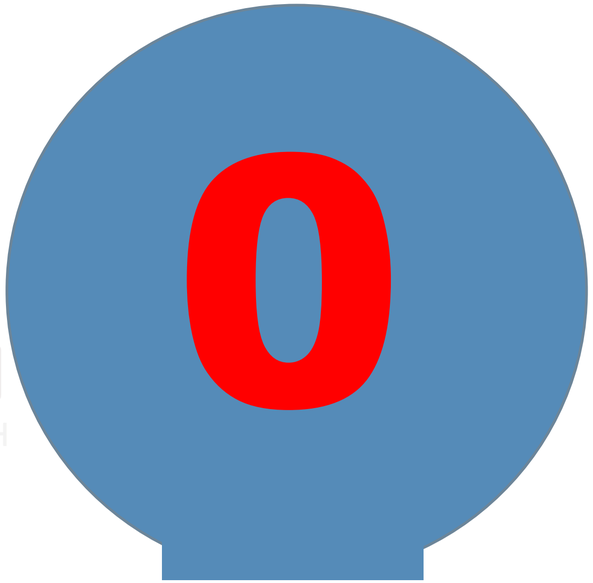
❾ Cho biết tổng <u>số phép gán</u> và <u>số phép so sánh</u> của các phần tử trong dãy

CYBERLEARN
ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH

Chèn trực tiếp
– Insertion sort

Đánh giá giải thuật

- Mỗi lần lặp:
  - ✓So sánh tìm vị trí thích hợp **pos**
  - ✓Dời chỗ phần tử sang phải 1 vị trí
- Giải thuật thực hiện tất cả N-1 vòng lặp



!!! Số lượng phép so sánh và dời chỗ này phụ thuộc vào tình trạng của dãy số ban đầu

CyberSoft.edu.vn - CyberLearn.vn

Chèn trực tiếp
– Insertion sort

Kết luận

- "Chèn trực tiếp" và "Chọn trực tiếp" đều có chi phí cho trường hợp xấu nhất là $O(n^2)$ do đó, không thích hợp cho việc sắp xếp các mảng lớn
- Dễ cài đặt, dễ kiểm lỗi
- "Chèn trực tiếp" tốt hơn "Chọn trực tiếp", nhất là khi mảng đã có thứ tự sẵn

❾ Cần có những giải thuật hiệu quả hơn cho việc sắp xếp các mảng lớn

CYBERLEARN
ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH

Quick sort

- Chia dãy cần sắp thành 2 phần
- Cách "chia": ½ dãy bên trái chứa các giá trị nhỏ hơn ½ dãy bên phải
- Thực hiện việc sắp xếp trên từng dãy con **(đệ qui)**

(x là phần tử trong dãy)

CYBERLEARN
ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH

Giải thuật phân hoạch dãy $a_L, a_{L+1}, \ldots a_R$ thành 2 dãy con

Bước 1.1:

Chọn tùy ý một phần tử a[k] trong dãy, L≤k≤R

x=a[k], i=L, j=R

Bước 1.2:

Phát hiện và hiệu chỉnh cặp a[i] và a[j] nằm sai chỗ:

Bước 1.2a: Trong khi (a[i]<x) i++

Bước 1.2b: Trong khi (a[j]>x) j--

Bước 1.2c: Nếu (i≤j): Hoán vị a[i] và a[j]; i++, j--

Bước 1.3:

Nếu i<j: Lặp lại bước 1.2

Ngược lại: Dừng phân hoạch

CYBERLEARN
ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH

CyberSoft.edu.vn - CyberLearn.vn

Quick sort

VD: 3; 5; 8; 10; 31; 4; 81; 7; 15; 17; 1.

Giả sử x = 10 thì sẽ có 2 phần như sau:

- Phần nhỏ hơn x: 3; 5; 8; 4; 7; 1
- Phần lớn hơn x: 31; 81; 15; 17

Giải thuật

Cho dãy $a_L, a_{L+1}, \ldots a_R$

Bước 1:

Phân hoạch dãy $a_L \ldots a_R$ thành các dãy con:

- Dãy con 1: $a_L \ldots a_j < x$
- Dãy con 2: $a_{j+1} \ldots a_{i-1} = x$
- Dãy con 3: $a_i \ldots a_R > x$

Bước 2:

- Nếu (L<j) Phân hoạch dãy $a_L \ldots a_j$
- Nếu (i<R) Phân hoạch dãy $a_i \ldots a_R$

CYBERLEARN
ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH

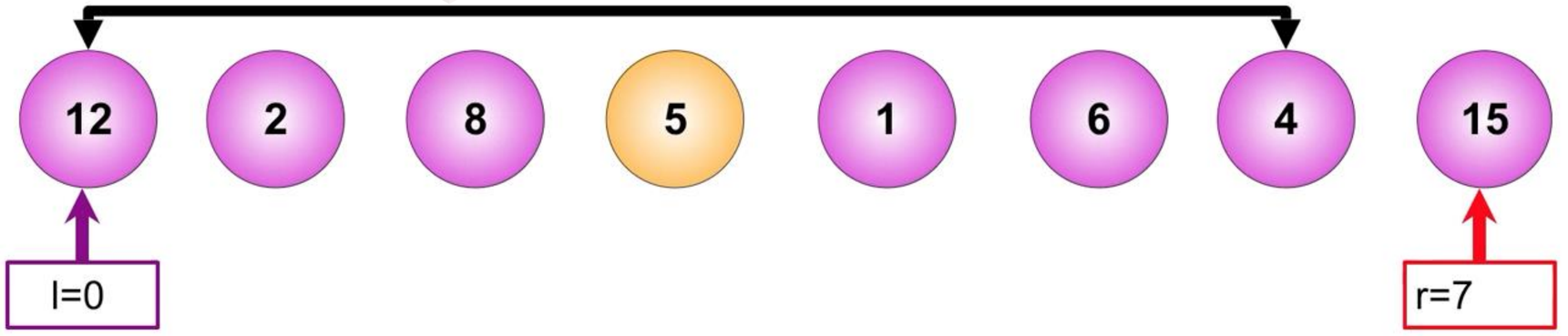
CyberSoft.edu.vn - CyberLearn.vn

Quick Sort – Ví Dụ

➢ Cho dãy số a:

12 2 8 5 1 6 4 15

Phân hoạch đoạn l =0, r = 7: x = a[3] = 5

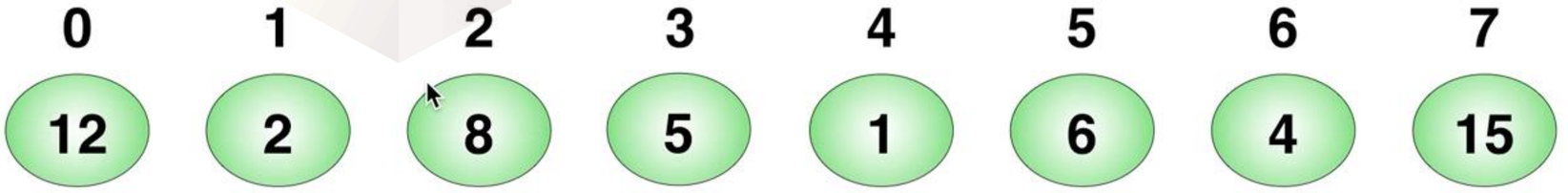


CYBERLEARN
ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH

CyberSoft.edu.vn - CyberLearn.vn

Quick Sort – Ví Dụ

➢ Phân hoạch đoạn [0,7]



CYBERLEARN
ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH

Đánh giá giải thuật

- Chi phí trung bình **$O(n*\log_2 n)$**
- Chi phí cho trường hợp xấu nhất **$O(n^2)$**
- Chi phí tùy thuộc vào cách chọn phần tử "trục":
  - Nếu chọn được phần tử có giá trị trung bình, ta sẽ chia thành 2 dãy bằng nhau;
  - Nếu chọn nhằm phần tử nhỏ nhất (hay lớn nhất) ➒ $O(n^2)$

CYBERLEARN
ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH

CyberSoft.edu.vn - CyberLearn.vn

Bài tập

Minh họa từng bước thực hiện của giải thuật Quick Sort khi sắp dãy số sau tăng dần:

15	7	9	10	6	20	6	9	12	30
1	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*

CYBERLEARN
ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH

Bài tập sắp xếp - Tự luyện

Cho nhập vào kích thước 2 mảng A và B. Tạo ngẫu nhiên giá trị cho các phần tử. Hãy

* Sắp xếp mảng A, B tăng dần bằng bất kì thuật toán sắp xếp nào.
* Trộn 2 mảng A, B đã sắp ở trên sao cho mảng kết quả là giảm dần. Không được sắp xếp trực tiếp trên mảng kết quả.

Ví dụ :

Nhập số phần tử cho mảng A và B (n, m > 0)

n = 5

m = 7

17 -21 0 100 -42

67 11 -66 32 52 22 -48

Mảng A sắp tăng: -42 -21 0 17 100

Mảng B sắp tăng: -66 -48 11 22 32 52 67

Trộn A và B thành C giảm dần:

100 67 52 32 22 17 11 0 -21 -42 -48 -66

CYBERLEARN
ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH