

Bài 9.3: Thuật toán sắp xếp chọn

- ✓ Các đặc điểm
- ✓ Thuật toán selection sort
- ✓ Ví dụ minh họa
- ✓ Bài tập thực hành

Các đặc điểm

- Là một thuật toán sắp xếp tập hợp có độ phức tạp $O(n^2)$.
- Thuật toán thực hiện việc sắp xếp bằng cách lần lượt tìm các phần tử nhỏ nhất từ phần chưa được sắp xếp đưa về phần đầu tập hợp.
- Thuật toán chia tập hợp(mảng) cần sắp xếp thành 2 phần:
 - Phần đầu mảng chứa các phần tử đã được sắp xếp.
 - Phần sau của mảng chứa các phần tử chưa được sắp xếp.
- Tại mỗi lần lặp, phần tử nhỏ nhất trong phần chưa được sắp xếp sẽ được đưa sang đúng vị trí của nó ở phần mảng đã được sắp xếp.

Thuật toán sắp xếp chọn

➤ Mã giả của thuật toán selection sort:

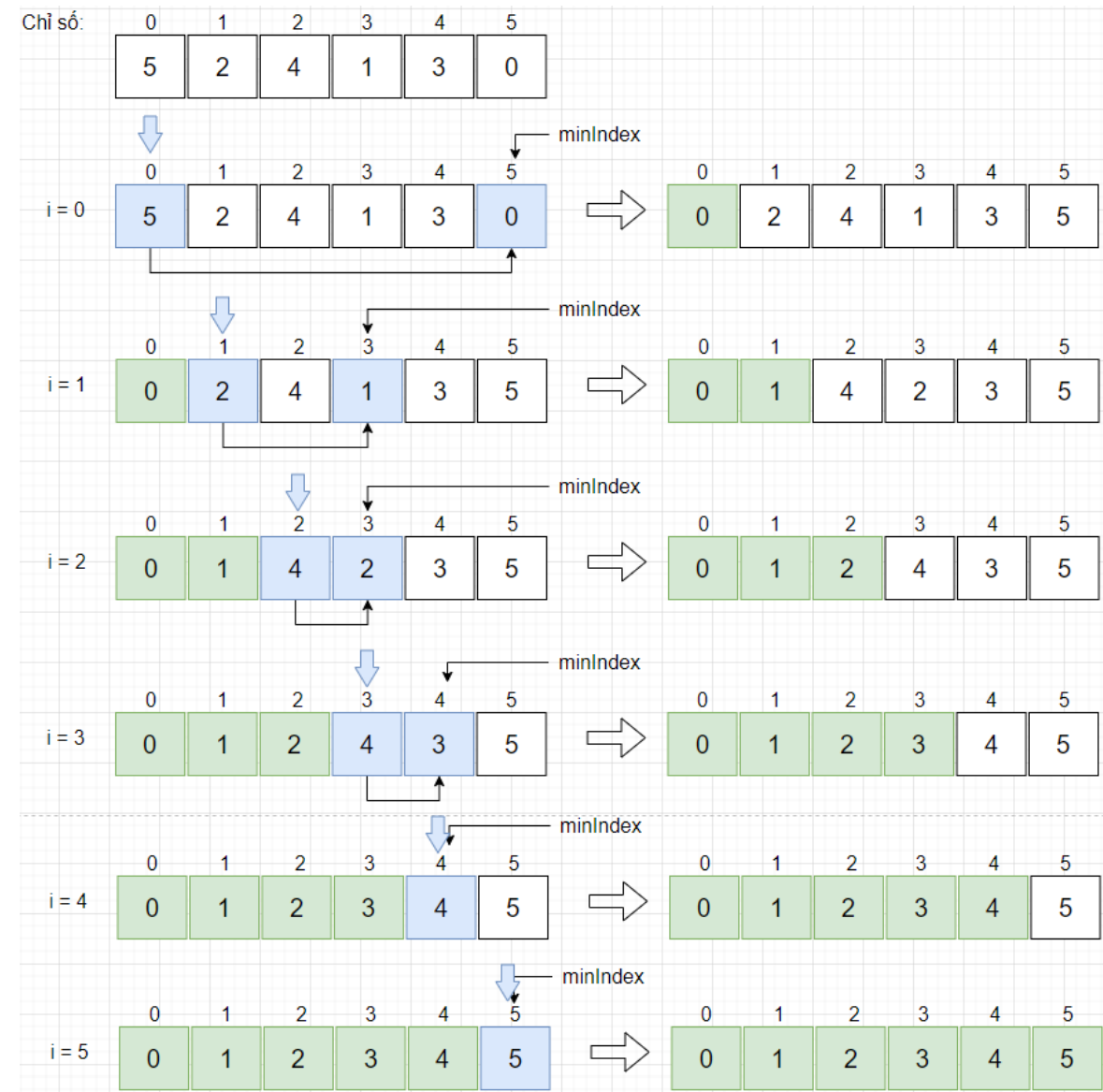
```
// thuật toán sắp xếp chọn
// arr: mảng đầu vào
// n: số phần tử của mảng
function selectionSort(arr[], n):
    for(i từ 0 đến n - 2):
        minIndex = i // gán minIndex là i
        for(j từ i + 1 đến n - 1):
            if(arr[j] < arr[minIndex]): // tìm thấy phần tử nhỏ hơn
                minIndex = j // cập nhật minIndex
        swap(arr[i], arr[minIndex]) // đổi chỗ hai phần tử
```

Thuật toán sắp xếp chọn

➤ Mã thật của thuật toán selection sort:

```
template<class T> void selectionSort(T* arr, size_t size) {  
    for (size_t i = 0; i < size - 1; i++)  
    {  
        size_t minIndex = i; // bắt đầu từ vị trí i  
        for (size_t j = i + 1; j < size; j++)  
        { // nếu phần tử tại vị trí j nhỏ hơn phần tử tại vị trí minIndex  
            if (arr[j] < arr[minIndex]) {  
                minIndex = j; // cập nhật minIndex  
            }  
        }  
        swap(arr[i], arr[minIndex]); // đổi chỗ hai phần tử tại vị trí i, minIndex  
    }  
}
```

Minh họa thuật toán



Nội dung tiếp theo

Thuật toán sắp xếp insertion sort