

COMPUTER SCIENCE PROGRAMMING C++

Lesson 11: SORT Sắp xếp

Nguyễn Văn Hiếu



FullHouse

Nội dung tìm hiểu



1

Sắp xếp là gì?
Tại sao lại cần sắp xếp?



2

Các thuật toán sắp xếp cơ bản



3

Hàm thuật toán sắp xếp có sẵn



4

Ứng dụng vào bài tập



01

INTRODUCTION

Sắp xếp là gì ?



FullHouse

C++

Sắp xếp là một giải thuật quan trọng trong máy tính

Sắp xếp là việc biến đổi dãy dữ liệu về một chiều tăng dần hoặc giảm dần






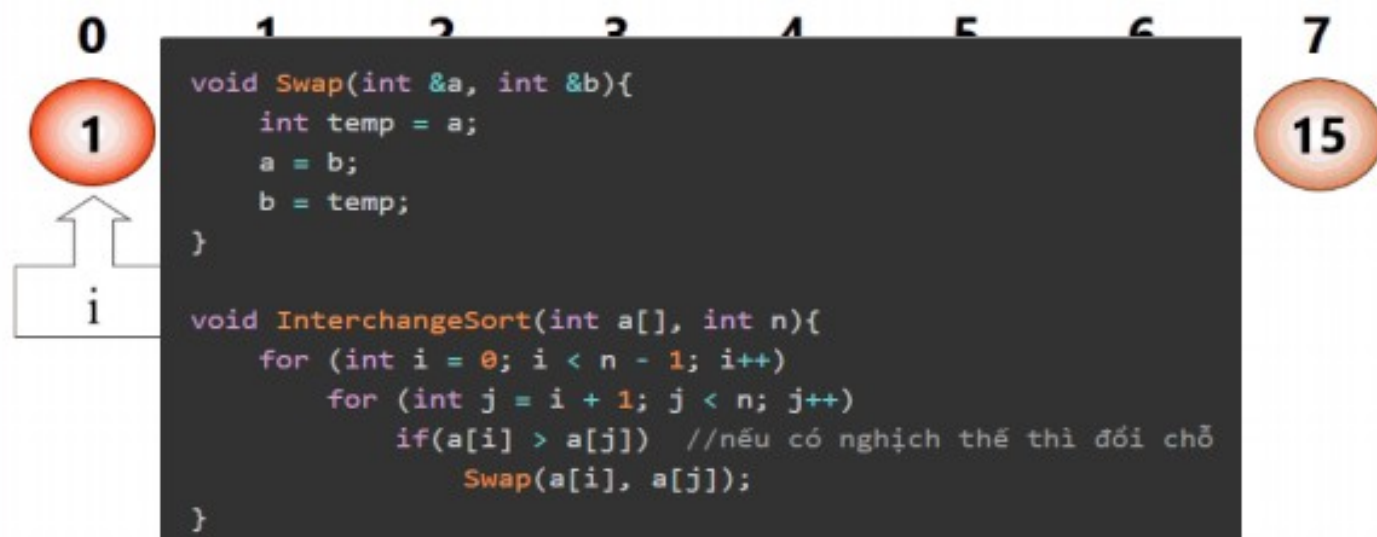
02

CÁC THUẬT TOÁN CƠ BẢN

Các thuật toán sắp xếp cơ bản



InterchangeSort



Nếu $a[i] > a[j]$ thì đổi chỗ $a[i], a[j]$



SelectionSort

```
void SelectionSort(int a[], int n)
{
    int min; // chỉ số phần tử nhỏ nhất trong dãy hiện hành
    for (int i = 0; i < n - 1; i++){
        min = i;
        for(int j = i + 1; j < n; j++){
            if (a[j] < a[min])
                min = j; // ghi nhận vị trí phần tử nhỏ nhất
        }
        if (min != i)
            Swap(a[min], a[i]);
    }
}
```

0

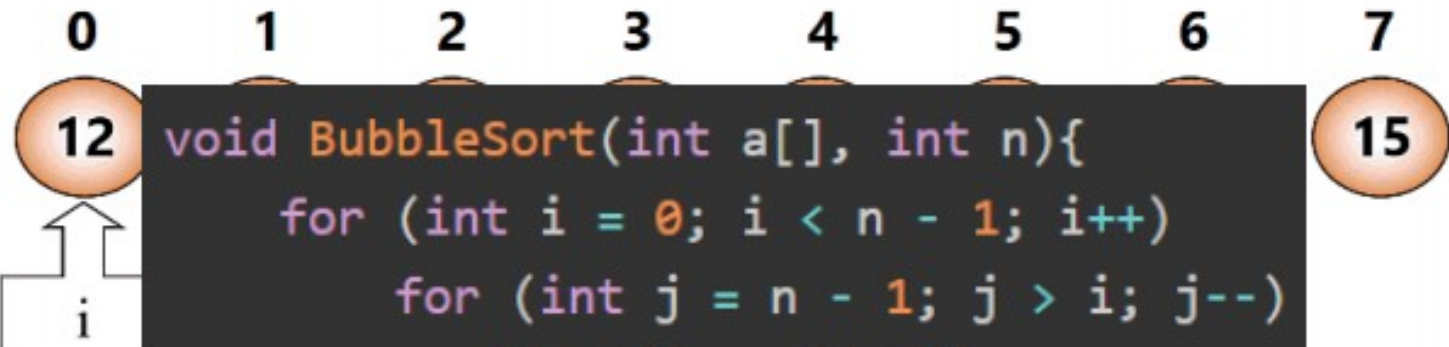
12

7

15



BubbleSort



```
void BubbleSort(int a[], int n){  
    for (int i = 0; i < n - 1; i++)  
        for (int j = n - 1; j > i; j--)  
            if(a[j] < a[j-1])  
                Swap(a[j], a[j-1]);  
}
```

Nếu $a[j] < a[j-1]$ thì đổi chỗ $a[j]$, $a[j-1]$

InsertSort

```
void InsertionSort(int a[], int n){  
    int pos, x;  
    for(int i = 1; i < n; i++){ //đoạn a[0] đã sắp  
        x = a[i];  
        pos = i;  
        while(pos > 0 && x < a[pos-1]){  
            a[pos] = a[pos-1]; // dời chỗ  
            pos--;  
        }  
        a[pos] = x;  
    }  
}
```

0

12

7

15



03

THUẬT TOÁN CÓ SẴN

Hàm sort có sẵn



Sort in algorithm

C++ là ngôn ngữ lập trình bậc trung và đã được hỗ trợ rất nhiều tiện ích

Bởi vì tích hợp sẵn mẫu hình OOP nên được cộng đồng support rất nhiều

Đối với việc sắp xếp, chúng ta có hàm thực hiện thuật toán đã được quy định sẵn

```
#include <iostream>
#include <vector>
#include <algorithm>
using namespace std;
int main()
{
    int a[6] = {5, 4, 3, 2, 1, 0};
    sort(a, a + 6);
    for (int i = 0; i < 6; ++i)
    {
        cout << a[i] << " ";
    }
    return 0;
}
```






04

COMPARATOR

Sử dụng comparator để set chiều



Comparator

Đôi khi sử dụng hàm sort không chỉ để sắp xếp tăng dần mà còn giảm dần

Sắp xếp từ giữa 2 vị trí cụ thể

Comparator ra đời để hỗ trợ chúng ta thực hiện điều đấy

```
#include <iostream>
#include <vector>
#include <algorithm>
using namespace std;
bool comparator(int a, int b)
{
    return a > b;
}
int main()
{
    vector<int> a = {1, 5, 9, 4, 7, 8, 2, 3, 6, 4, 0};
    sort(a.begin(), a.end(), comparator);
    for (int i = 0; i < a.size(); ++i)
    {
        cout << a.at(i) << " ";
    }
    return 0;
}
```



The background features a dark blue field with glowing yellow circuit-like lines. These lines form a grid-like pattern with various branches and junctions. Small yellow squares are placed at several points along these lines, particularly on the left and right sides, resembling components on a circuit board.

C++

Ứng dụng giải quyết bài tập



Full House

The background features a complex network of glowing blue lines that resemble a circuit board. These lines include straight paths, right-angle turns, and loops. Several small blue squares and circles are placed at various points along these lines, acting as nodes or components. The overall aesthetic is high-tech and digital.

Thank

Cảm ơn tất cả các bạn đã theo dõi



FullHouse