

Bài 1: Giới thiệu về cấu trúc dữ liệu và giải thuật

(Introduction to data structures and algorithms)



Lê Sỹ Vinh
Bộ Môn Khoa Học Máy Tính – Khoa CNTT
Đại Học Công Nghệ - ĐHQGHN
Email: vinhioi@yahoo.com

Cấu trúc dữ liệu (data structure)

- Cấu trúc dữ liệu là gì?
Cấu trúc dữ liệu là cách tổ chức lưu giữ dữ liệu trong sao cho hiệu quả nhất
- Thế nào là hiệu quả?
 1. **“Chính xác”**
 2. Dùng ít bộ nhớ
 3. Khả năng tìm kiếm/truy xuất
 4. Khả năng cập nhật, thêm bớt (modification, insertion / deletion)
 5. Đơn giản, dễ hiểu
- Các kiểu cấu trúc dữ liệu cơ bản
 - Bản ghi (struct)
 - Danh sách (array)
 - Danh sách liên kết (list)
 - Cây (tree)
 - Bảng băm (hash table)

Thuật toán (algorithm)

- Thuật toán là gì?

Thuật toán là một phương pháp bao gồm một dãy các bước tính toán để giải quyết một bài toán. Thuật toán có thể được diễn tả dưới dạng ngôn ngữ tự nhiên (tiếng Việt, tiếng Anh...) hay ngôn ngữ lập trình (C++, Java...)

- Thế nào là một thuật toán tốt?

1. “Đúng đắn”
2. Nhanh
3. Ít bộ nhớ
4. Đơn giản, dễ hiểu

Ví dụ 1: Sắp xếp danh sách tuyển sinh

Năm 2008, Đại học Công Nghệ có N thí sinh tham gia tuyển sinh, hãy viết chương trình sắp xếp các thí sinh theo thứ tự giảm dần của tổng điểm thi ba môn

Ví dụ:

Stt	Họ tên	Toán	Lý	Hóa	Tổng
1	Trần Anh Tuấn	7	8	7	22
2	Bùi Ngọc Thăng	10	10	9	29
3	Lê Sỹ Vinh	10	8	8	26
4	Nguyễn Thị Ánh	8	10	9	27

Sắp xếp nổi bọt (bubble sort)

Ý tưởng: Lần lượt duyệt qua danh sách thí sinh, nếu hai thí sinh không đúng thứ tự, đổi chỗ hai thí sinh. Lặp lại quá trình trên cho đến khi danh sách được sắp xếp

Step 0

1. (Tuấn, 22)
2. (Thăng, 29)
3. (Vinh, 26)
4. (Ánh, 27)

Step 1

1. (Thăng, 29)
2. (Tuấn, 22)
3. (Vinh, 26)
4. (Ánh, 27)

Step 3

1. (Thăng, 29)
2. (Vinh, 26)
3. (Tuấn, 22)
4. (Ánh, 27)

Step 4

1. (Thăng, 29)
2. (Vinh, 26)
3. (Ánh, 27)
4. (Tuấn, 22)

Step 5

1. (Thăng, 29)
2. (Ánh, 27)
3. (Vinh, 26)
4. (Tuấn, 22)

Sắp xếp nổi bọt (bubble sort)

```
Function bubbleSort ( $A$  : danh sách thí sinh) {  
    swapped := false;  
    do  
        swapped := false;  
        for each  $i = 1$  to  $N - 1$  do  
            if  $A[i].diem < A[i + 1].diem$  then {  
                swap ( $A[i], A[i+1]$ );  
                swapped := true;  
            }  
        done;  
    while (swapped = true)  
}
```

Vietnam - Google Search - Mozilla Firefox

File Edit View History Bookmarks Tools Help


http://www.google.com/search?hl=en&q=+Vietnam&btnG=Search

Getting Started HbVar ID 2504 Latest Headlines

Web Images Maps News Shopping Gmail more Sign in

Google Vietnam Search Advanced Search Preferences

Web Maps Blogs News Images Books Video Groups Results 1 - 10 of about 280,000,000 for Vietnam [definition]. (0.18 seconds)

 [Vietnam maps.google.com](https://maps.google.com)

[Vietnam - Wikipedia, the free encyclopedia](#)
Vietnam (pronounced /vi-ət-ˈnɑːm/; Vietnamese: **Việt Nam**), officially the Socialist Republic of Vietnam (Vietnamese: Cộng hòa xã hội chủ nghĩa **Việt Nam**; ...
en.wikipedia.org/wiki/Vietnam - 276k - [Cached](#) - [Similar pages](#)

[Vietnam War - Wikipedia, the free encyclopedia](#)
The war was fought between the communist North Vietnam, supported by its communist allies, and South Vietnam, supported by the United States and others. ...
en.wikipedia.org/wiki/Vietnam_War - 345k - [Cached](#) - [Similar pages](#)
[More results from en.wikipedia.org »](#)

[Vietnam Travel Guide and Travel Information - Lonely Planet](#)
Lonely Planet Vietnam Travel Guide. Vietnam travel information, advice, hotels, reviews, maps and itineraries.
www.lonelyplanet.com/worldguide/vietnam/ - 54k - [Cached](#) - [Similar pages](#)

[Vietnam Travel](#)
Vietnam travel and information, including the best airfare and hotels on Vietnam.com.
www.vietnam.com/ - 37k - [Cached](#) - [Similar pages](#)

[Tong cuc Du lich Viet Nam - Vietnam National Administration of Tourism](#)
Official Website of Vietnam National Administration of Tourism, includes information about Vietnam people and country and tourism.
www.vietnamtourism.com/ - 7k - [Cached](#) - [Similar pages](#)

[CIA - The World Factbook -- Vietnam](#)
Features map and brief descriptions of the geography, people, government, economy, communications, transportation, military and transnational issues.
<https://www.cia.gov/library/publications/the-world-factbook/geos/vn.html> - 118k - [Cached](#) - [Similar pages](#)

[American Experience | Vietnam Online | PBS](#)
Vietnam Online: In a landmark PBS series and an award-winning companion Website, Done

Done

Vietnam - Google S...

9:32 AM

Ví dụ 1': Sắp xếp danh sách website (google search)

Google có danh sách N website. Website x có một độ ưu tiên là $f(x)$. Hãy sắp xếp các website trên theo độ ưu tiên giảm dần

Câu hỏi: Có thể dùng bubble sort không?

Trả lời: Được, **nhưng không hiệu quả**

Ví dụ 2: Danh bạ điện thoại

Viết một chương trình quản lý danh bạ điện thoại của toàn bộ thành phố Hà Nội, sao cho các thao tác sau được hiệu quả nhất:

1. Kiểm tra một số điện thoại
2. Thêm một số điện thoại
3. Xóa một số điện thoại

Ví dụ 3: Tìm đường đi tốt nhất

- Xây dựng hệ thống phần mềm chỉ đường đi tốt nhất cho người dùng
 1. Đường đi ngắn nhất
 2. Đường đi qua ít đèn xanh – đèn đỏ nhất
 3. Đường đi ít tắc nhất

Ví dụ 3: Tìm đường đi tốt nhất (google map)

W 172nd St, New York, NY 10032 to 106 W 83rd St, New York, NY 10024 - Google Maps - Mozilla Firefox

File Edit View History Bookmarks Tools Help

http://maps.google.com/

google map Search Bookmarks Check AutoLink AutoFill Send to google map Settings

Web Images Maps News Shopping Gmail more Sign in | Help

Google Maps New York Search Maps Show search options Find businesses, addresses and places of interest. Learn more.

Get Directions My Maps

Print Send Link

Street View Traffic More... Map Satellite Terrain

A 172nd street 10032 New York
B 81nd street 10024 new york
Add Destination - Show options
Walking Get Directions

Also available: By car

Walking directions are in beta.
Use caution - This route may be missing sidewalks or pedestrian paths.

Walking directions to 106 W 83rd St, New York, NY 10024
4.8 mi - about 1 hour 38 mins

A W 172nd St
New York, NY 10032

1. Head southeast on W 172nd St toward St Nicholas Ave 164 ft
2. Turn right at St Nicholas Ave 0.1 mi
3. Continue on Broadway 4.3 mi
4. Turn left at W 83rd St 0.2 mi

B 106 West 83rd LLC
106 W 83rd St
New York, NY 10024

Ví dụ 3: Tìm đường đi tốt nhất (google map)

Washington, DC to New York, NY - Google Maps - Mozilla Firefox

File Edit View History Bookmarks Tools Help

http://maps.google.com/

google map

Web Images Maps News Shopping Gmail more

Sign in | Help

Google Maps

New York

Search Maps Show search options

Find businesses, addresses and places of interest. Learn more.

Get Directions My Maps

Print Send Link

Map Satellite Terrain

Washington DC

New York

Add Destination - Show options

By car

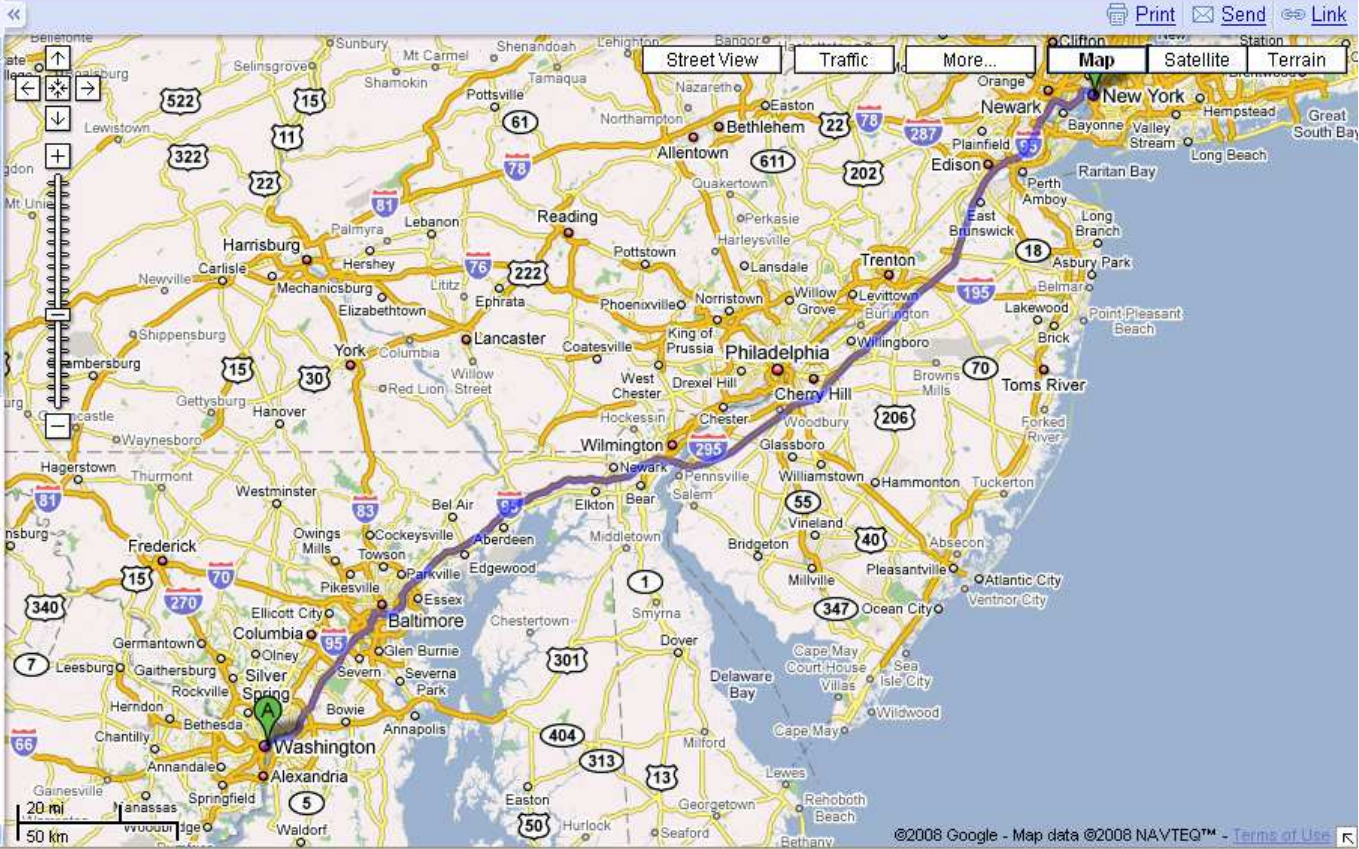
Get Directions

Driving directions to New York, NY

227 mi – about 4 hours 10 mins (up to 4 hours 50 mins in traffic)

Washington, DC

1. Head north on 14th St NW/US-1 toward Madison Dr NW 0.1 mi
2. Turn right at Constitution Ave NW/US-1 0.6 mi
3. Turn left at 6th St NW/US-1 0.8 mi
4. Turn right at New York Ave NW/US-50 E Entering Maryland 4.6 mi
5. Take the Balt- Wash. Pkwy exit on the left toward Baltimore 0.5 mi
6. Merge onto MD-295 N 29.8 mi
7. Take the exit onto I-95 N toward New York Partial toll road Entering Delaware 67.8 mi
8. Take the I-295 exit toward New Jersey 0.2 mi



©2008 Google - Map data ©2008 NAVTEQ™ - Terms of Use

Done

Ví dụ 4: Xây dựng hệ thống từ điển

Viết chương trình từ điển Anh – Việt, cho phép thực hiện các thao tác sau:

1. Tìm một từ
2. Thêm một từ
3. Xóa một từ
4. Sửa một từ
5. Tìm từ đồng nghĩa

Ví dụ 5: Người bán hàng traveling salesman problem (TSP)

Một người bán hàng cần đến thăm N khách hàng ở N địa điểm khác nhau. Tìm một hành trình cho người bán hàng trên sao cho:

1. Mỗi địa điểm thăm đúng 1 lần, sau đó quay về điểm xuất phát
2. Tổng chi phí đi lại là ít nhất



Người bán hàng

Thuật toán: Thăm địa điểm gần nhất (nearest neighbor tour)

Từ điểm xuất phát, lần lượt đi thăm các điểm theo quy tắc: “Đến thăm điểm chưa được thăm gần với điểm hiện tại nhất”



Người bán hàng



Nearest neighbor tour: 1 → 2 → 3 → X → 7 → 8 → 6 → 5 → 4 → 9 → 1

Đường đi tối ưu: 1 → 2 → 3 → 4 → 5 → 6 → 8 → 7 → X → 9 → 1

Các ví dụ khác (10')

Thế nào là một chương trình tốt?

1. Đúng đắn
2. Hiệu quả
3. Dễ hiểu
4. Dễ tìm lỗi
5. Dễ thay đổi và nâng cấp

“Thuật toán + Cấu trúc dữ liệu = Chương trình”

N. Wirth

Dữ liệu

- Dữ liệu là những thông tin mà máy tính có thể xử lý: số nguyên, số thực, xâu kí tự, và các dữ liệu phức tạp được tạo từ nhiều thành phần
- Trong bộ nhớ máy tính, dữ liệu được biểu diễn dưới dạng nhị phân (dãy các kí tự 0, 1)
- Trong các ngôn ngữ lập trình bậc cao (C++, Java..), dữ liệu được biểu diễn dưới dạng trừu tượng, xuất phát từ biểu diễn toán học và dễ hiểu cho con người:
 - int age
 - double weight

Kiểu dữ liệu cơ bản

Kiểu dữ liệu được xác định bởi:

1. Phạm vi giá trị
2. Các phép toán

Ví dụ trong C++

kiểu	phạm vi	phép toán thường dùng
➤ bool	true / false	and, or, not
➤ char	-127 -> 127	'<', '>', '='
➤ int	-32,767 -> 32,767	'<', '>', '=', '+', '-', '*', '/'
➤ float	~1E-37 -> ~1E+37	'<', '>', '=', '+', '-', '*', '/'
➤ double	~1.7E-308 -> ~1.7E+308	'<', '>', '=', '+', '-', '*', '/'

Kiểu dữ liệu có cấu trúc

Câu hỏi: Làm sao để biểu diễn dữ liệu về 1 điểm trên mặt phẳng?

Đáp án: Ngôn ngữ lập trình cung cấp cho ta những luật để xây dựng kiểu dữ liệu mới T từ những kiểu dữ liệu đã biết t_1, t_2, \dots, t_n .

Ví dụ trong C++:

```
struct  $T$  {  
     $t_1$   $x_1$   
     $t_2$   $x_2$   
    .....  
     $t_n$   $x_n$   
}
```

Kiểu dữ liệu có cấu trúc

- Xây dựng cấu trúc dữ liệu để biểu diễn dữ liệu của 1 điểm trên mặt phẳng

```
struct pointType {  
    double    x;  
    double    y;  
}
```

- Xây dựng cấu trúc dữ liệu để biểu diễn dữ liệu của 1 đoạn thẳng trên mặt phẳng

```
struct lineType {  
    point Type    start;  
    pointType     end;  
}
```

Kiểu dữ liệu có cấu trúc

- Xây dựng cấu trúc dữ liệu để biểu diễn 1 sinh viên (5')

```
struct studentType {  
    char    name[100];  
    int     age;  
    bool    sex;  
}
```

- Xây dựng cấu trúc dữ liệu để biểu diễn danh sách 1 lớp học

```
struct studentClassType{  
    char            className[100];  
    int            numberStudent;  
    studentType    studentArr[100];  
}
```

Phạm vi và các phép toán trên kiểu dữ liệu có cấu trúc

Xét kiểu dữ liệu mới T được tạo từ những kiểu dữ liệu đã biết t_1, t_2, \dots, t_n ,

Ví dụ:

```
struct complexType {  
    double real;  
    double image;  
}
```

Phạm vi: Xác định bởi phạm vi của các kiểu dữ liệu thành phần

- real: là số thực nằm trong phạm vi kiểu ‘double’
- image: là số thực nằm trong phạm vi kiểu ‘double’

Phạm vi và các phép toán trên kiểu dữ liệu có cấu trúc

Phép toán: Do người dùng định nghĩa

Ví dụ:

```
struct complexType {  
    double  real;  
    double  image;  
}
```

```
complexType createComplex (double real, double image) {  
    complexType c;  
    c.real = real;  
    c.image = image;  
    return c;  
}
```

Phạm vi và các phép toán trên kiểu dữ liệu có cấu trúc

```
complexType add (complexType c1, complexType c2) {  
    complexType c12;  
    c12.real = c1.real + c2.real;  
    c12.image = c1.image + c2.image;  
    return c12;  
}
```

```
complexType multiply (complexType c1, complexType c2) {  
    complexType c12;  
    c12.real = (c1.real * c2.real) - (c1.image * c2.image);  
    c12.image = (c1.real * c2.image) + (c1.image * c2.real);  
    return c12;  
}
```

Phạm vi và các phép toán trên kiểu dữ liệu có cấu trúc

```
complexType getReal (complexType c) {  
    c.real  
}
```

```
complexType getImage (complexType c) {  
    c.image  
}
```

```
void printComplex (complexType c) {  
    cout << c.real << “ +i ” << c.image << “ \n” ;  
}
```

Trừu tượng hóa dữ liệu

(abstraction data type)

1. Đặc tả đối tượng dữ liệu (các thành phần dữ liệu của đối tượng)

Ví dụ: đối tượng số phức (complex)

- real
- image

2. Đặc tả các phép toán trên đối tượng dữ liệu (operations)

Ví dụ: Đối tượng số phức (complex):

- createComplex (real, image)
- getReal (complexNumber)
- getImage (complexNumber)
- add (complexNumber1, complexNumber2)
- multiply (complexNumber2, complexNumber2)
- print (complexNumber)

Trừu tượng hóa dữ liệu

Trừu tượng hóa đối tượng sinh viên (student) (5’)

1. Đặc tả đối tượng dữ liệu

name, age, sex, address

2. Đặc tả các phép toán trên đối tượng dữ liệu

createStudent (name, age, sex, address)

compare (student1, student2)

getName (student)

getAge (student)

getSex (student)

getAdd (student)

Trừu tượng hóa dữ liệu

- studentClass

1. Đặc tả đối tượng dữ liệu

className, numberStudent, studentArr, Address

2. Đặc tả các phép toán trên đối tượng dữ liệu

addStudent (studentClass, student)

findStudent (studentClass, student)

deleteStudent (studentClass, student)

getClassName (studentClass)

getNumberStudent (studentClass)

getStudentArr (studentClass)

getStudentAddress (studentClass)

Lập trình hướng đối tượng

Object oriented programming (OOP)

- Lập trình hướng đối tượng giúp chúng ta cài đặt các mô tả trừu tượng (đối tượng dữ liệu và các phép toán) thành các đoạn mã chương trình
- Chương trình được thiết kế thành từng đoạn nhỏ, mỗi đoạn mô tả về một đối tượng (thuộc tính dữ liệu, các phép toán trên dữ liệu)
- Hai thuộc tính quan trọng: đóng gói (encapsulation) và thừa kế (inheritance)

OOP: Tính đóng gói (encapsulation)

- **Class:** Cài đặt một lớp đối tượng dữ liệu trừu tượng. Việc cài đặt bao gồm cài đặt các thành phần dữ liệu và các phép toán trên dữ liệu

Ví dụ:

```
class complex {  
    private:  
        double real;  
        double image;  
  
    public:  
        void create (double newReal, double newImage) {  
            real = newReal; image = newImage;  
        }  
        double getReal () {  
            return real;  
        }  
        .....  
        void print {  
            cout << real << “ +i ” << image << “ \n” ;  
        }  
};
```

- Liên kết chặt chẽ giữa dữ liệu và phép toán
- Che dấu dữ liệu
- Dễ dàng tìm lỗi
- Các đối tượng liên kết với nhau thông qua các phép toán

OOP: Tính đóng gói (encapsulation)

Object: Biểu diễn cho một đối tượng cụ thể của một lớp

complex c1;

complex c2;

Thiết kế chương trình

- Đặc tả vấn đề
- Thiết kế cấu trúc dữ liệu và giải thuật
- Cài đặt (C++, Java...)
- Thử nghiệm và sửa lỗi

Thiết kế chương trình: Đặc tả vấn đề

Chính xác hóa vấn đề cần giải quyết:

- Chúng ta được cho những gì?
- Chúng ta cần tìm ra cái gì?
- Mối quan hệ giữa chúng là gì?

Đặc tả vấn đề trong khoa học máy tính:

Input: Dữ liệu vào, các ràng buộc, *định dạng*

Output: Dữ liệu ra, các ràng buộc, *định dạng*

Đặc tả vấn đề

Ví dụ: Cho một dãy số phức, hãy

1. Tính tổng của dãy số phức
2. Tính tích của dãy số phức
3. Tìm số phức có phần thực (real) lớn nhất
4. Tìm số phức có phần ảo (image) lớn nhất

Đặc tả vấn đề:

- Input: Một dãy số phức, mỗi số phức được biểu diễn bởi 2 số thực mô tả phần thực (real) và phần ảo (image)
- Output:
 - c1 (số phức biểu diễn tổng của dãy số phức)
 - c2 (số phức biểu diễn tích của dãy số phức)
 - c3 (số phức có phần thực lớn nhất)
 - c4 (số phức có phần ảo lớn nhất)

Bài tập

Đặc tả vấn đề cho các bài toán dưới đây

1. Sắp xếp danh sách website
2. Hệ thống từ điển
3. Tìm đường đi tốt nhất
4. Người bán hàng