

Giới thiệu về Cấu trúc dữ liệu và Giải thuật

Bùi Ngọc Thắng

September 8, 2015

Outline

- 1 Mục tiêu
- 2 Cấu trúc dữ liệu
- 3 Giải thuật
- 4 Bài tập

Mục tiêu bài học

- Tìm hiểu về khái niệm cấu trúc dữ liệu
- Tìm hiểu về các loại cấu trúc dữ liệu
- Tìm hiểu về khái niệm thuật toán

Cấu trúc dữ liệu – Khái niệm

Trong Khoa học máy tính, **cấu trúc dữ liệu** là một cách **tổ chức dữ liệu trong máy tính** nhằm mục đích sử dụng dữ liệu một cách hiệu quả.

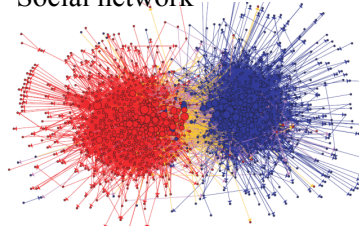
A portion of the DNA sequence with
length of 1,6 million characters

```
...  
TACATTAGTTATTACATTGAGAACTTTATAATTAAAA  
AGATTCATGTAAATTTCTTATTTGTTTATTAGAGGTT  
TTAAATTTAATTTCTAAGGGTTTGCTGGTTTCATTGTT  
AGAATATTAACTTAATCAAATATTTGAATTAAATTAG  
GATTAATTAGGTAAGCTAACAAATAAGTTAAATTTTAA  
ATTTAAGGAGATAAAAATACTACTCTGTTTTATTATGG  
AAAGAAAGATTTAAATACTAAAGGGTTTATATATATGAA  
GTAGTTACCCTTAGAAAAATATGGTATAGAAAGCTTAA  
ATATTAAGAGTGATGAAGTATATTATGT...
```

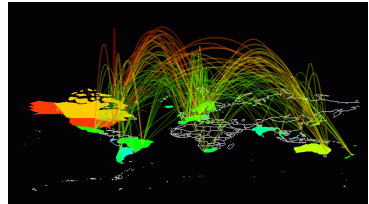
Immense text



Social network



Web linkage



Dữ liệu thô cần thiết phải tìm cách để biểu diễn dữ liệu phù hợp
trong máy tính → tìm kiếm các đặc trưng quan trọng của dữ liệu
để biểu diễn chúng

Cấu trúc dữ liệu – Các loại cấu trúc dữ liệu

Dữ liệu thô → tiền xử lý dữ liệu (loại bỏ dữ liệu lỗi, nhiễu, ..) → lựa chọn/trích chọn đặc trưng để biểu diễn dữ liệu.

- Các kiểu dữ liệu (data types)
- Cấu trúc dữ liệu có thứ tự (ordered data structures, linear data structures)
- Cấu trúc cây (tree)
- Cấu trúc bảng băm (hashing table)
- Cấu trúc đồ thị (Graph)

Cấu trúc dữ liệu – Kiểu dữ liệu

- Dữ liệu phạm trù (categorical data)
 - Đúng sai (boolean). Ex: `bool var1;`
 - Ký tự (Character). Ex: `char var1;`
 - Xâu ký tự (String). Ex: `string var1;`
- Dữ liệu số (numerical data)
 - Số nguyên (Integer). Ex: `int var1, var2;`
 - Số thực (Float). Ex: `float var1; double var2;`

Cấu trúc dữ liệu – Kiểu dữ liệu (tiếp)

- Kiểu dữ liệu mảng (Array).
Ex: `int Arr[10][10];`
- Kiểu dữ liệu trừu tượng (struct, class).
Ex: `Box b1;`

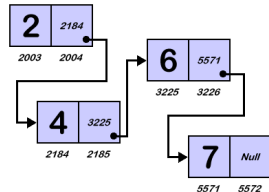
```
class Box {  
public:  
    double length;  
    double width ;  
    double height;  
};
```



Cấu trúc dữ liệu – Cấu trúc dữ liệu có thứ tự

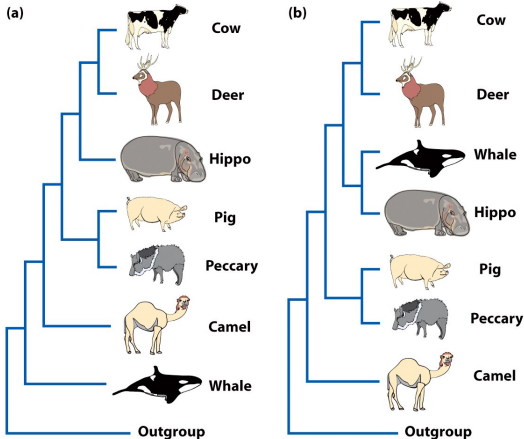
Các đối tượng dữ liệu được liệt kê theo thứ tự

- Mảng
- danh sách và danh sách liên kết (List and linked list)



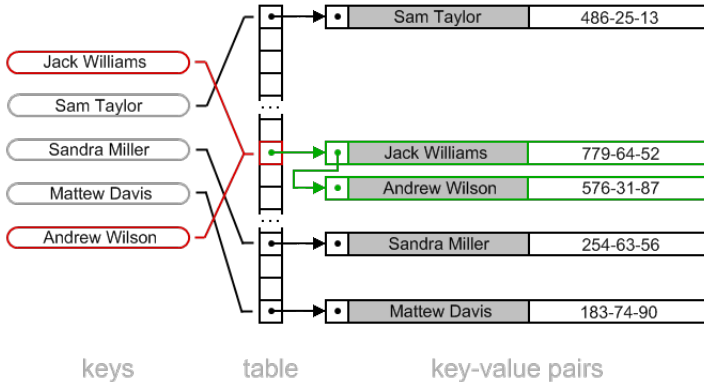
Cấu trúc dữ liệu – Cấu trúc cây

Các đối tượng dữ liệu được lưu trữ dưới dạng cây (bao gồm: gốc, nút trong, lá và mối quan hệ giữa các nút)

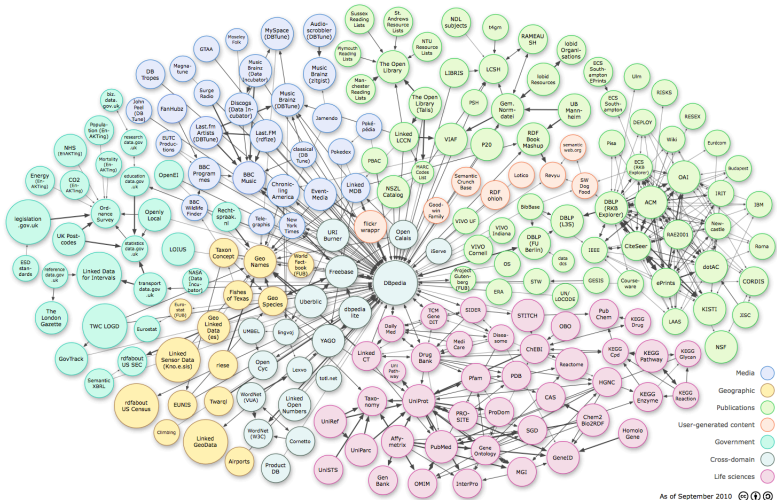


Cấu trúc dữ liệu – Bảng băm

Lưu trữ dữ liệu bằng cách sử dụng một hàm ánh xạ (băm) khóa của dữ liệu tới một giá trị xác định



Cấu trúc dữ liệu – Đồ thị



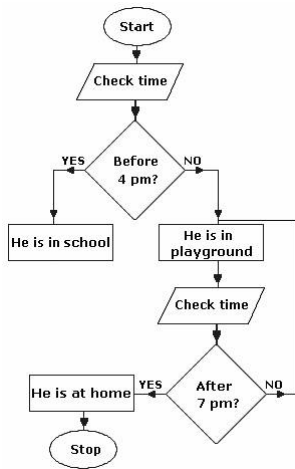
Thuật toán – Khái niệm

- 1 Là một niệm quan trọng trong tin học. Thuật toán là **một dãy hữu hạn các bước**, mỗi bước **mô tả chính xác** các phép toán hoặc các hành động cần thực hiện để giải quyết một vấn đề.
- 2 Là một **tập các quy tắc** để giải quyết các vấn đề trong một số **hữu hạn các bước**.

Các phương pháp mô tả thuật toán

- Sử dụng ngôn ngữ tự nhiên (tiếng anh hoặc tiếng việt,..)
- Kết hợp ngôn ngữ tự nhiên và ngôn ngữ toán học
- Sử dụng sơ đồ khối
- Kết hợp ngôn ngữ tự nhiên, ngôn ngữ toán học và ngôn ngữ lập trình
- Ngôn ngữ lập trình

Thuật toán – Sơ đồ khối và mã giả



INSERTION-SORT(A)

```
1 for  $j = 2$  to  $A.length$ 
2    $key = A[j]$ 
3   // Insert  $A[j]$  into the sorted
   sequence  $A[1..j-1]$ .
4    $i = j - 1$ 
5   while  $i > 0$  and  $A[i] > key$ 
6      $A[i + 1] = A[i]$ 
7      $i = i - 1$ 
8    $A[i + 1] = key$ 
```

Thuật toán – Các đặc trưng của thuật toán

① **Input:**

- Xác định nguồn dữ liệu.
- Biểu diễn dữ liệu (cấu trúc dữ liệu)

② **Output:** Là thông tin, dữ liệu mong muốn đạt được khi giải quyết vấn đề trên dữ liệu đầu vào (input) có sẵn.

③ **Tính xác định:** Mỗi bước cần mô tả một cách chính xác, chỉ có một cách hiểu duy nhất.

④ **Tính khả thi:** Tất cả các phép toán có mặt trong các bước của thuật toán phải đủ đơn giản.

⑤ **Tính dừng:** Sau một số bước hữu hạn thuật toán phải dừng và đưa ra kết quả.

Thuật toán – Các vấn đề quan tâm

- 1 Thiết kế hoặc lựa chọn thuật toán phù hợp.
- 2 Thuật toán sử dụng phải đúng đắn.
- 3 Thuật toán sử dụng phải hiệu quả: đơn giản, dễ hiểu, dễ cài đặt, độ phức tạp chấp nhận được (càng nhanh càng tốt, càng ít tài nguyên càng tốt).

Xác định dữ liệu đầu vào, dữ liệu ra, và xây dựng tư tưởng giải thuật cho các bài toán sau đây

- 1 Kiểm tra một số nguyên N có phải là số hoàn thiện không
- 2 Tìm giá trị lớn nhất của một dãy số thực
- 3 Tính $n!$
- 4 Nhập vào một dãy gồm N phần tử, kiểm tra dãy có phải là dãy tăng không.
- 5 Nhập vào một dãy gồm N phần tử, sắp xếp dãy theo thứ tự tăng dần.
- 6 Tìm ước chung lớn nhất của hai số a và b .