Tên: Ngô Thị Kim Tài

MSSV:2051050420

**CÂU HỎI CHƯƠNG 1**

**Câu 1: Trong khoa học máy tính, cấu trúc dữ liệu được hiểu như thế nào? Cho ví dụ.**

* CTDL là cấu trúc (sự tổ chức) của dữ liệu/thông tin lên trên máy tính, mà ở đó với cấu trúc này máy tính có thể xử lý được.
* Cấu trúc này phải rõ ràng, xác định, các thành phần bên trong cấu trúc cũng phải rõ ràng, và xác định.

**VD:** Thông tin cơ bản của một sinh viên: Mã số sinh viên, họ và tên, giới tính, ngày sinh và địa chỉ. Trong đó:

+ MSSV, Họ tên và địa chỉ có kiểu dữ liệu là kiểu chuỗi.

+ Ngày sinh có kiểu Date.

**Câu 2: Trong khoa học máy tính, giải thuật được hiểu như thế nào? Cho ví dụ.**

* Giải thuật là một tập hữu hạn của các bước (chỉ thị hay hành động) theo một trình tự, được xác định rõ ràng nhằm mục đích để giải quyết một bài toán nào đó (dựa vào những giá trị đầu vào gọi là “input” và cho ra kết quả đầu ra gọi là “ouput”).

**VD:** Cho 3 số nguyên a, b, c .Tìm số lớn nhất trong ba số đã cho.

**- Input:** 3 số nguyên a, b, c.

**- Output:** số lớn nhất trong ba số.

Thuật toán:

Bước 1: Gán max := a.

Bước 2: Nếu max < b thì gán max := b.

Bước 3: Nếu max < c thì gán max := c.

Bước 4: Output max.

* Tư tưởng của thuật toán lệ duyệt lần lượt giá trị của từng số và giữ lại giá trị lớn nhất vào biến max.
* Kết thúc thuật toán max cho số nguyên lớn nhất trong 3 số đã cho.

**Câu 3: Tại sao nói CTDL và GT có quan hệ mật thiêt với nhau? Liệt kê 1 ví dụ nói về cách thiết kế cấu trúc dữ liệu sẽ ảnh hưởng đến giải thuật, giải thích tại sao?**

* Cấu trúc dữ liệu và giải thuật có mối quan hệ mật thiết với nhau bởi vì nó được thể hiện qua công thức:

**Cấu trúc dữ liệu + Giải thuật = Chương trình**

* Với một cấu trúc dữ liệu đã chọn sẽ có những giải thuật tương ứng, phù hợp.
* Khi cấu trúc dữ liệu thay đổi thường giải thuật cũng phải thay đổi theo để tránh việc xử lý gượng ép, thiếu tự nhiên trên một cấu trúc không phù hợp.
* Cấu trúc dữ liệu tốt sẽ giúp giải thuật xử lý trên đó có thể phát huy tác dụng tốt hơn, vừa đáp ứng nhanh vừa tiết kiệm bộ nhớ, giải thuật cũng dễ hiểu và đơn giản hơn.

**VD:** Giả sử ta có danh sách gồm những cặp “Tên đơn vị, Số điện thoại”: (a1,b1), (a2,b2).....(an,bn).

Ta muốn viết một chương trình cho máy tính điện tử để khi cho biết “tên đơn vị” máy sẽ in ra cho ta: “Số điện thoại”. Đó là một loại bài toán mà phép xử lý cơ bản là “tìm kiếm”.

* Một cách đơn giản là cứ đếm lần lượt các tên trong danh sách a1,a2... cho tới khi tìm thấy tên đơn vị ai nào đó, đã chỉ định, thì đối chiếu ra số điện thoại bi của nó. Nhưng việc đó chỉ làm được khi danh mục điện thoại ngắn, nghĩa là với n nhỏ, còn với n lớn thì rất mất thời gian.
* Nếu trước đó danh mục điện thoại đã được sắp xếp theo thứ tự từ điển đối với tên đơn vị, tất nhiên sẽ áp dụng một giải thuật tìm kiếm khác tốt hơn như ta thường làm khi tra từ điển.
* Nếu lại tổ chức thêm một bảng mục lục chỉ dẫn theo chữ cái đầu tiên của “Tên đơn vị” , chắc rằng khi tìm số điện thoại của Đại Học Mở Tp.HCM ta sẽ bỏ qua được các tên đơn vị mà chữ đầu không phải là “Đ” -> **Thời gian tìm kiếm nhanh hơn.**

Như vậy: giữa cấu trúc dữ liệu và giải thuật có mối quan hệ mật thiết. Có thể coi chúng như hình với bóng. Không thể nói tới cái này mà không nhắc tới cái kia.

**Câu 4: Đếm số phép so sánh trong giải thuật ở ví dụ 1.12.**

* i chạy so sánh ở vòng for được n lần
* j chạy tổng n lần
* a[j – 1] chạy tổng n – 1 lần
* tổng : n+n+ n – 1 = 3n - 1