**BÀI TẬP LÝ THUYẾT CHƯƠNG I**

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

Cấu trúc dư liệu (sự tổ chức) của dữ liệu / Thông tin 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.

Ví dụ: Cấu trúc dữ liệu cơ bản của một kho hàng (Tên hàng, mã hàng, giá tiền, tồn kho)

Trong đó:

-Mã hàng, tên hàng có kiểu dữ liệu là chuỗi.

Gía tiền, tồn kho có kiểu dữ liệu là số thực.

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

Giaỉ 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 kết quả đầu ra “Output”

Ví dụ: Tìm X trong phương trình ax + b = 0 (\*)

Gỉai thuật

Đầu vào (input): a, b.

Đầu ra (output): Kết luận nghiệm

-Bước 1: Chuyển b qua vế phải: (ax = -b)

-Bước 2: Chia cả hai vế cho a: (x = -b/a)

-Bước 3: Kết luận phương trình (\*) có nghiệm duy nhất x = -b/a.

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

Giaỉ thuật: Là một hệ thống chặt chẽ và rõ ràng các qui tắc nhằm xác định một dãy các thao tác trên những đối tượng, sao cho sau một số bước hữu hạn thực hiện các thao tác đó ta thu được kết quả mong muốn.

Cấu trúc dữ liệu: Là cách tổ chức, lưu trữ dữ liệu trong MTDL một cách có thứ tự, có hệ thống nhằm sử dụng dữ liệu một cách hiệu quả, CTDL và GT có mối liên hệ chặt chẽ với nhau, chúng luôn tồn tại song song đi kèm nhau theo công thức: Cấu trúc dữ liệu + Chương trình = Giải thuật.

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

Số phép so sánh trong ví dụ 1.12: Là 3(n).

**BÀI TẬP THỰC HÀNH CHƯƠNG I**

Bài 1: Đếm số phép toán gán, phép so sánh được thực thi và xác định độ phức tạp trong đoạn code sau:

For(i=0; i<n; i++)

For(j=0; j<m; j++)

If(a[i][j] == x)

Return 1;

Return -1;

-Số phép toán gán: O(n)

-Số phép so sánh: O(n)

-Độ phức tạp: O(n^2)

Bài 2: Đếm số phép toán gán, phép so sánh được thực thi và xác định độ phức tạp trong đoạn code sau:

Sum=0;

For( i=0; i<n; i++)

For( j=0; j<I; j++)

Sum++;

-Số phép toán gán: O(n)

-Số phép so sánh: O(n)

-Độ phức tạp: O(n^2)

Bài 3: Đánh giá độ phức tạp của đoạn code sau:

For( i=0; i<n; i++)

Sum1 +=I;

For( i=0; i<n\*n; i++)

Sum2 +=I;

-Độ phức tạp: O(n)

Bài 4: Đánh giá độ phức tạp của hàm tính giai thừa sau:

Int GT(int n)

{

If( n <= 1)

Return 1;

Return n\*GT( n-1);

}

-Độ phức tạp: O( n^2)

Bài 5: Đánh giá độ phức tạp của hàm tính dãy Fibonacci sau:

Int Fibo(int n)

{

If( n <= 1)

Return n;

Return Fibo(n-1) + Fibo(n-2);

}

-Độ phức tạp: O(n^2).