**Chương 1:**

**CẤU TRÚC DỮ LIỆU VÀ GIẢI THUẬT**

**CÂU HỎI**

Câu 1:

- Trong khoa học máy tính, cấu trúc dữ liệu là hệ thống tổ chức sắp xếp dữ liệu trong đó việc lưu trữ và cách lưu trữ dữ liệu sao cho việc truy xuất và xử lí dữ liệu đó đạt hiệu quả cao nhất.

- Ví dụ:

Cấu trúc dữ liệu cơ bản của một Nhân Viên gồm Mã Nhân Viên, họ tên, giới tính, năm sinh. Trong đó:

+ MSSV, họ tên, giới tính có kiểu dữ liệu là kiểu chuỗi. (kiểu char)

+ Năm sinh của nhân viên có kiểu Số Nguyên. (kiểu int)

Câu 2:

- Trong khoa học máy tính, giải thuật là một quy trình được định nghĩa và tính toán kĩ lưỡng, với đầu vào là giá trị nào đó hoặc tập các giá trị và trả ra kết quả cũng là giá trị nào đó hoặc tập các giá trị, hay còn gọi là đầu ra. Hay nói cách khác, giải thuật chính là tập hợp tuần tự các bước tính toán biến đổi đầu vào thành đầu ra.

- Ví dụ:

Tìm Số Nhỏ Nhất trong một mảng các Phần Tử.

Bước 1: Tạo 1 biến min bằng giá trị phần tử đầu tiên.

Bước 2: Duyệt từ phần tử thứ 2 đến cuối danh sách.

Bước 3: So sánh min với các phần tử còn lại trong mảng. Nếu có phần tử nhỏ hơn min thì ta đặt min bằng phần tử đó.

Câu 3:

- Nói CTDL và GT có quan hệ mật thiết với nhau vì khi giải quyết một bài toán trên máy tính, để xác định được giải thuật phù hợp cần phải biết nó tác động đến loại dữ liệu nào và khi chọn lựa cấu trúc dữ liệu cũng cần phải hiểu rõ những thao tác nào nào sẽ tác động đến nó. Do vậy, GT và CTDL có mối quan hệ chặt chẽ với nhau được thể hiện qua công thức:

CTDL + GT = Chương trình

\*Ví dụ: Để tính số buổi vắng mặt của Nhân Viên, người ta thường dùng biến kiểu Số Nguyên để tính tổng thay vì để kiểu Chuỗi sẽ khó để tính tổng.

Câu 4:

- Vòng while:

Với j = 0: có 1 phép so sánh

Với j = 1: có 3 phép so sánh

Với j = 2: có 5 phép so sánh

Với j = 3: có 7 phép so sánh

Với j = k: có (2k + 1) phép so sánh

=> T(n) = (2n + 1)

- Vòng for:

Với n = 0: có 1 phép so sánh

Với n = 1: có 2 phép so sánh

Với n = 2: có 3 phép so sánh

Với n = k: có (k + 1) phép so sánh

=> T(n) = (n + 1)

T = (n + 1)(2n + 1) thuộc lớp O()