**Bài 1:**

* Trong khoa học máy tính, **cấu trúc dữ liệu** được hiểu 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 lý** dữ liệu đó đạt hiệu quả cao nhất.

Ví dụ: dữ liệu danh sách về điểm của sinh viên có dạng sinh viên môn 1, môn 2, môn 3 ….

**Bài 2:**

* Là một quy trình được định nghĩa và tính toán kỹ 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, thuật toán chính là tập hợp tuần tự các bước tính toán, biến đổi đầu vào thành đầu ra.

Ví dụ: nhập vào phương trình bật 2 và xuất ra gia trị X cần tìm.

**Bài 3:**

* Thực hiện bài toán thực tế thành bài toán có thể giải quyết trên máy tính, để xây dựng một mô hình tin học phản ánh được bài toán thực tế cần :

+ tổ chức , xây dựng các cấu trúc thích hợp nhất sao cho vừa có thể phản ánh chính xác các dữ liệu thực tế này, vừa có thể dễ dàng dùng máy tính để xử lý. Công việc này được gọi là xây dựng cấu trúc dữ liệu cho bài toán.

+ Từ những yêu cầu xử lý thực tế, cần tìm ra các giải thuật tương ứng để xác định trình tự các thao tác máy tính phải thi hành để cho ra kết quả mong muốn, đây là bước xây dựng giải thuật cho bài toán.

+ Tuy nhiên khi giải quyết một bài toán trên máy tính, chúng ta thường có khuynh hướng chỉ chú trọng đến việc xây dựng giải thuật mà quên đi tầm quan trọng của việc tổ chức dữ liệu trong bài toán. Giải thuật phản ánh các phép xử lý , còn đối tượng xử lý của giải thuật lại là dữ liệu, chính dữ liệu chứa đựng các thông tin cần thiết để thực hiện giải thuật. Để xác định được giải thuật phù hợp cần phải biết nó tác động đến loại dữ liệu nào (ví dụ để làm nhuyễn các hạt đậu , người ta dùng cách xay chứ không băm bằng dao, vì đậu sẽ văng ra ngoài) và khi chọn lựa cấu trúc dữ liệu cũng cần phải hiểu rõ những thao tác nào sẽ tác động đến nó (ví dụ để biểu diễn các điểm số của sinh viên người ta dùng số thực thay vì chuỗi ký tự vì còn phải thực hiện thao tác tính trung bình từ những điểm số đó). Như vậy trong một đề án tin học, giải thuật và cấu trúc dữ liệu có mối quan hệ chặt chẽ với nhau, 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. Hơn nữa, một 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 vật tư, giải thuật cũng dễ hiễu và đơn giản hơn.

Ví dụ : Một chương trình quản lý điểm thi của sinh viên cần lưu trữ các điểm số của 3 sinh viên. Do mỗi sinh viên có 4 điểm số ứng với 4 môn học khác nhau nên dữ liệu có dạng bảng như sau:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Sinh viên | Môn 1 | Môn 2 | Môn 3 | Môn 4 |
| Sv1 | 7 | 9 | 5 | 2 |
| Sv2 | 5 | 0 | 9 | 4 |
| Sv3 | 6 | 3 | 7 | 4 |

Chỉ xét thao tác xử lý là xuất điểm số các môn của từng sinh viên.

Giả sử có các phương án tổ chức lưu trữ sau:

Phương án 1 : Sử dụng mảng một chiều.

Phương án 2 : Sử dụng mảng 2 chiều.

Suy ra, Có thể thấy rõ phương án 2 cung cấp một cấu trúc lưu trữ phù hợp với dữ liệu thực tế hơn phương án 1, và do vậy giải thuật xử lý trên cấu trúc dữ liệu của phương án 2 cũng đơn giản, tự nhiên hơn.

**Bài 4:**

Mỗi lần duyệt, ta luôn phải hoán vị 1 lần, 1 lần hoán vị tương đương 3 phép gán, nghĩa là thuật toán sẽ tốn 3\*(n-1) + 3\*(n-2) +…+ 3 = 3n\*(n-1)/2=O(n2) phép gán.