**Họ và tên:** Nguyễn Hoàng Minh Thư

**MSSV:** 1851050142

**Nhóm:** 11

**Chương 1**

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

Cấu trúc dữ liệu 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, là cách lưu trữ, tổ chức dữ liệu có thứ tự, có hệ thống, phải rõ ràng, xác định các thành phần bên trong cấu trúc cũng phải rõ ràng, và xác định.

VD: Cấu trúc cơ bản của một hàng hóa (mã hàng, tên hàng, xuất xứ, số lượng)

* Trong đó:

+ Mã hàng, tên hàng, xuất xứ có kiểu dữ liệu là kiểu chuỗi.’

+ Số lượng có kiểu dữ liệu kiểu int (kiểu số nguyên).

**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 “**output”**

VD: Tìm kết quả cho phép tính cộng 2 số:

Bước 1: Khai báo 3 số: a, b, c

Bước 2: Định nghĩa giá trị của a & b

Bước 3: Cộng các giá trị của a & b

Bước 4: Lưu trữ kết quả này vào biến c

Bước 5: In biến c. Kết thúc

**Câu 3: Tại sao nói cấu trúc dữ liệu và giải thuật có quan hệ mật thiết 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?**

Giải thuật chỉ phản ánh các phép xử lý, còn đối tượng để xử lý trên máy tính điện tử, chính là dữ liệu chúng biểu diễn các thông tin cần thiết cho bài toán: các dữ kiện đưa vào, các kết quả trung gian… Không thể nói tới giải thuật mà không nghĩ tới: giải thuật đó được tác động trên dữ liệu nào, còn khi xét tới dữ liệu thì cũng phải hiểu: dữ liệu ấy cần được tác động giải thuật gì để đưa tới kết quả mong muốn. Với một cấu trúc dữ liệu sẽ có một thuật giải tương ứng. Cấu trúc dữ liệu thay đổi, giải thuật cũng thay đổi theo.

VD: Giả sử có các cặp danh sách gồm “mã số sinh viên, họ tên sinh viên”:

* Tìm lần lượt các mã số theo một đơn vị nào đó cho tới khi tìm được đối tượng cần tìm thấy, đối chiếu để lấy họ tên sinh viên. Nhưng việc đó chỉ làm được khi danh sách đó ít, nếu không sẽ rất mất thời gian.
* Sắp xếp các danh sách theo một thứ tự từ số nhỏ nhất cho tới lớn và phân đoạn.

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

Với i = 1 thì j chạy phép so sánh là 2 lần

Với i = 2 thì j chạy phép so sánh là 4 lần

.

.

.

Với i = n thì j chạy phép so sánh là n lần

* T(n) = n\*n O(n2)

**BÀI TẬP\_Trả lời**

**Bài 1:**

|  |  |
| --- | --- |
| **Phép gán** | |
| + m = 0 thì có 1 phép gán  + m = 1 thì có 2 phép gán  .  .  .  + m thì có m + 1 phép gán | + n = 0 thì có m + 1 (for m) \*0 + 1 gán  + n = 1 thì có m + 1 (for m) \*1 + 1 gán  .  .  .  + n thì có m + 1 (for m)\*n + 1 gán |
| * T(m,n) = (m + 1) \* n + 1 * Nếu m = n thì T(n) = (n + 1)\*n + 1   O(n2) | |

|  |  |
| --- | --- |
| **Phép so sánh** | |
| + m = 0 thì có 1 phép so sánh  + m = 1 thì có 3 phép so sánh  .  .  .  + m thì có m + 2 phép so sánh | + n = 0 thì có (2m + 1) \* 0 + 1 so sánh  + n = 1 thì có (2m + 1) \* 1 + 1 so sánh  .  .  .  + n thì có (2m + 1) \* n + 1 so sánh |
| * T(m,n) = (2m + 1)\*n + 1 * Nếu m = n thì T(n) = (2n + 1) \* n + 1   O(n2) | |

**Bài 2:**

* **Phép gán**

+ n = 0 thì i có 1 phép gán

+ n = 1 thì i có 2 phép gán, j có 1 phép gán

+ n = 2 thì i có 3 phép gán, j có 2 phép gán

….

+ n thì i có n + 1 phép gán, j có n phép gán

* T(n) O(n2)
* Phép so sánh tương tự phép gán

**Bài 3:**

for (i = 0; i < n; i++) ⬄O(n)

sum1+=i;

for (i = 0; i < n\*n; i++) ⬄O(n)

sum2+=i;

**Vậy:** T(n) = max( O(n), O(n) ) O(n2)

**Bài 4:**

T(n) = C1 (khi n = 1)

T(n) = T(n – 1) + C2 (khi n > 1)

T(n) = T (n – 1) + C2

⬄ T(n) = (T(n – 2) + C2) + C2 = T(n – 2) + 2C2

….

⬄(T(n – k) + C2) + (k – 1)C2

Chương trình dừng khi n – k = 1 => k = n – 1

⬄T(n – n – 1) + (n – 1)C2 = T(1) + (n + 1)C2

⬄ C1 + (n – 1)C2

**Vậy:** T(n) O(n)

**Bài 5:**

T(n) = C1 (khi n = 0 và n = 1)

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

Ta có Fibonaci tăng dần theo mũ => T(n) = O(nk)