Câu 1 : Cơ chế lưu trữ biến của lập trình C

**Biến cục bộ (Local variables):** Biến được khai báo bên trong một hàm hoặc khối mã. Khi hàm kết thúc, bộ nhớ được giải phóng và biến cục bộ không còn tồn tại nữa.

**Biến toàn cục (Global variables):** Biến được khai báo ngoài tất cả các hàm và có thể truy cập từ bất kỳ đâu trong chương trình. Biến toàn cục được lưu trữ trong vùng nhớ được cấp phát khi chương trình bắt đầu và chỉ bị giải phóng khi chương trình kết thúc.

**Biến tĩnh (Static variables):** Biến tĩnh được khai báo bằng từ khóa static. Các biến tĩnh được lưu trữ trong vùng dữ liệu tĩnh và tồn tại suốt thời gian chạy của chương trình. Biến tĩnh cục bộ được khởi tạo một lần và giữ giá trị giữa các lần gọi của hàm mà chúng thuộc về.

**Con trỏ (Pointers):** Con trỏ lưu trữ địa chỉ bộ nhớ của một biến khác. Chúng cho phép truy cập trực tiếp đến vùng nhớ và thay đổi giá trị của biến đó.

**Vùng nhớ động (Dynamic memory):** Trong C, có các hàm như malloc, calloc, và realloc trong thư viện stdlib.h cho phép cấp phát vùng nhớ động. Vùng nhớ này được cấp phát và giải phóng theo yêu cầu của chương trình.

Câu 2 : Toán tử 3 ngôi trong C

Toán tử ba ngôi trong ngôn ngữ lập trình C được gọi là toán tử điều kiện (conditional operator), thường còn được gọi là toán tử điều kiện ba ngôi (ternary operator).

Cú pháp : (condition) ? expression1 : expression2;

Với nếu condition true => trả về expression1 và nếu false thì trả về expression2

Ex: int a = 20;

Int b = 10;

Int max = (a>b) ? a: b; // nếu a > b , max = a và ngược lại

Toán tử ba ngôi thường được sử dụng để rút ngắn việc viết các mệnh đề if-else trong C và cung cấp cách viết gọn hơn trong trường hợp cần so sánh giữa hai giá trị hoặc thực hiện các phép gán dựa trên một điều kiện.

Câu 3 : Dữ liệu bool và cách sử dụng

Dữ liệu boolean đề cập đến một kiểu dữ liệu có thể có giá trị True hoặc False. Dữ liệu boolean thường được sử dụng trong lập trình để kiểm tra điều kiện. Nó là một kiểu dữ liệu cơ bản trong nhiều ngôn ngữ lập trình.

Cách dùng : Đặt tên biến và gắn giá trị True or False

Câu 4 : Tại sao đang nhập số nhưng khi chuyển qua kí tự thì không được và sao phải dùng fflush stdin

- Đây là một vấn đề thường gặp khi sử dụng hàm scanf để nhập số và ký tự trong C. Nguyên nhân là do hàm scanf chỉ đọc dữ liệu từ bàn phím cho đến khi gặp ký tự khoảng trắng, dấu chấm phẩy, hoặc dấu xuống dòng. Khi nhập số, hàm scanf sẽ để lại dấu xuống dòng trong bộ nhớ đệm, và khi nhập ký tự, hàm scanf sẽ đọc dấu xuống dòng đó và bỏ qua việc nhập ký tự.

- Cách khắc phục là dùng hàm getchar để đọc và xóa dấu xuống dòng trong bộ nhớ đệm trước khi nhập ký tự

- Hàm fflush stdin dùng để xóa bộ nhớ đệm, tránh dữ liệu cũ trong bộ nhớ đệm ảnh hưởng đến dữ liệu mới.