**I. CÁC LỖI TRUY XUẤT DỮ LIỆU (DATA REFERENCE ERRORS)**

**1. Dùng biến chưa có giá trị xác định ?**

int i, count;

for (i = 0; i < count; i++) {...}

**Trả lời:** Lỗi ở đây là biến count chưa được khởi tạo giá trị nào trước khi sử dụng trong vòng lặp. Điều này có thể dẫn đến kết quả không xác định hoặc vòng lặp không thể được thực hiện đúng.

**2. Dùng chỉ số của biến array nằm ngoài phạm vi?**

int list[10];

if (list[10] == 0) {...}

**Trả lời:** Lỗi ở đây là bạn đang truy cập phần tử thứ 11 của mảng list có kích thước chỉ là 10. Chỉ số của mảng bắt đầu từ 0 đến 9. Việc truy cập ngoài phạm vi có thể dẫn đến lỗi đọc dữ liệu không xác định hoặc ghi đè vào vùng nhớ khác.

**3. Dùng chỉ số không thuộc kiểu nguyên của biến array ?**

**int list[10];**

double idx=3.1416;

if (list[idx] == 0) {...}

**Trả lời:** Lỗi ở đây là bạn đang sử dụng một biến kiểu double làm chỉ số của mảng list. Chỉ số của mảng phải là kiểu nguyên. Việc sử dụng một giá trị không phải kiểu nguyên có thể dẫn đến kết quả không xác định hoặc lỗi truy cập.

**II - Các lỗi khai báo dữ liệu**

**1. Tất cả các biến đều được định nghĩa hay khai báo tường minh chưa?**

int i;

extern double d;

d = i\*10;

**Trả lời:** Lỗi ở đây là bạn đang thực hiện phép gán d = i\*10; ngay sau khai báo, trong khi biến d được khai báo là một biến extern (bên ngoài). Biến extern là một tham chiếu đến biến được định nghĩa ở nơi khác trong mã chương trình. Bạn không thể gán giá trị cho một biến extern như cách bạn đang làm.

**2. Định nghĩa hay khai báo đầy đủ các thuộc tính của biến dữ liệu chưa?**

static int i = 10;

**Trả lời:** Khai báo biến i với thuộc tính static và khởi tạo giá trị ban đầu là 10.

**4. Kiểu và độ dài từng biến đã được định nghĩa đúng theo yêu cầu chưa ?**

short IPAddress;

byte Port;

**Trả lời:** Lỗi ở đây là kiểu dữ liệu byte không được hỗ trợ bởi ngôn ngữ lập trình C chuẩn. Bạn cần sử dụng kiểu dữ liệu phù hợp khác, chẳng hạn unsigned char.

**3. Biến array hay biến chuỗi được khởi động đúng chưa ?**

int pi[10] = {1,5,7,9} ;

**Trả lời:** Lỗi ở đây là bạn khởi tạo mảng pi với một phần tử khởi tạo có giá trị là 1, phần tử thứ 2 có giá trị là 5, phần tử thứ 3 có giá trị là 7 và phần tử thứ 4 có giá trị là 9. Các phần tử còn lại sẽ có giá trị mặc định là 0. Bạn nên cung cấp đủ giá trị khởi tạo cho mảng hoặc để tất cả các phần tử mặc định bằng 0 nếu không cần khởi tạo.

**III. Các lỗi tính toán (Computation Errors)**

**1.**`CString` có vẻ giống như một loại chuỗi ký tự, và phép toán chia (`/`) không được định nghĩa trực tiếp cho chuỗi ký tự

**2.** Thực hiện phép nhân giữa một số nguyên (int) và một số thực (double), sau đó gán kết quả cho một biến kiểu byte. Vì các kiểu dữ liệu này không tương thích trực tiếp, nó có thể gây ra các vấn đề liên quan đến kiểu dữ liệu và mất dữ liệu

**3.** thực hiện phép nhân giữa một số nguyên (int) và một hằng số nguyên (500), sau đó gán kết quả cho một biến kiểu byte, có một số điều cần lưu ý. Kiểu byte thường biểu diễn một dải giá trị nhất định, ví dụ như từ 0 đến 255. Nếu kết quả của phép nhân i \* 500 vượt qua khoảng giá trị này, có thể xảy ra lỗi hoặc mất dữ liệu

**IV. Các lỗi so sánh (Comparison Errors)**

**1.** So sánh hai dữ liệu có kiểu không tương thích có thể dẫn đến kết quả không mong muốn và không đáng tin cậy. Trong ví dụ, ta đang cố gắng so sánh một biến kiểu int và một mảng ký tự (char[]), điều này không được hỗ trợ trong hầu hết các ngôn ngữ lập trình và có thể gây ra hiện tượng không mong muốn

**2.** Trong trường hợp này, ta đang so sánh một biến kiểu int và một biến kiểu char, hai kiểu dữ liệu không cùng độ dài. So sánh trực tiếp giữa các kiểu không cùng độ dài thường không phải là một hành vi mong muốn trong hầu hết các ngôn ngữ lập trình, vì nó có thể dẫn đến các vấn đề không đáng tin cậy.

**3.** Có hai vấn đề liên quan đến việc sử dụng các toán tử so sánh:

a.Độ ưu tiên và sự lộn xộn giữa các toán tử: Đôi khi, việc sử dụng nhiều toán tử so sánh trong một biểu thức có thể dẫn đến sự lộn xộn về mặt độ ưu tiên của các toán tử. Điều này có thể làm cho mã của bạn khó hiểu và gây hiểu nhầm.

b.Toán tử ngữ nghĩa tương tự: Toán tử `<=` và `>=` cùng với `and` và `or` có thể dễ dàng bị lẫn lộn vì chúng có ý nghĩa tương tự với các toán tử `less than or equal to`, `greater than or equal to`, `logical AND` và `logical OR`.

**V- Các lỗi luồng điều khiển (Control-Flow Errors)**

**1. Thiếu thực hiện 1 số nhánh trong lệnh quyết định theo điều kiện số học? switch (i) {**

**case 1: ... //cần hay không cần lệnh break;**

**case 2: ...**

**case 3: ...**

**}**

**Trả lời:** Cần, Trong lệnh switch, nếu bạn thiếu lệnh break sau mỗi case, một hiện tượng gọi là "fall-through" sẽ xảy ra. Điều này có nghĩa là sau khi một case được thực hiện, các case phía sau của nó cũng sẽ được thực hiện mà không cần kiểm tra điều kiện của chúng.

**2.Mỗi vòng lặp thực hiện ít nhất 1 lần hay sẽ kết thúc?**

**for (i=x ; i<=z; i++) {...} //nếu x > z ngay từ đầu thì sao?**

**for (i = 1; i <= 10; i--) {...} //có dừng được không?**

**Trả lời:** Trong trường hợp của vòng lặp for (i = x; i <= z; i++) {...}, nếu x lớn hơn z ngay từ đầu, điều kiện i <= z sẽ sai ngay từ đầu và vòng lặp sẽ không thực hiện bất kỳ lần lặp nào. Vòng lặp sẽ kết thúc ngay từ bước ban đầu.

Trong trường hợp của vòng lặp for (i = 1; i <= 10; i--) {...}, điều kiện i <= 10 sẽ luôn sai vì i đang giảm dần và không bao giờ lớn hơn hoặc bằng 10. Do đó, vòng lặp sẽ không bao giờ thực hiện và sẽ kết thúc ngay từ đầu.

**3.Biên của vòng lặp có bị lệch? for (i = 0; i <= 10; i++) {...} //hay i < 10 ?**

**Trả lời:** Trong trường hợp của vòng lặp for (i = 0; i <= 10; i++) {...}, biên trên của vòng lặp là i <= 10. Điều này có nghĩa là vòng lặp sẽ thực hiện khi i nhỏ hơn hoặc bằng 10. Vòng lặp sẽ thực hiện cho các giá trị của i từ 0 đến 10 bao gồm cả 10. Do đó, không có sự lệch nào xảy ra ở đây.

**4. Có đủ và đúng cặp token begin/end, {} ?**

**Trả lời**: Có đủ và đúng cặp token begin/end, {}.

**VI- Các lỗi giao tiếp (Interface Errors)**

1. **Số lượng tham số cụ thể được truyền có bằng số tham số hình thức của hàm**

**được gọi?**

**Trả lời:**  **Số lượng tham số**: Lỗi này xảy ra khi số lượng tham số mà bạn truyền vào hàm không khớp với số lượng tham số hình thức mà hàm đó yêu cầu. Điều này có thể gây ra lỗi thực thi và không đáng tin cậy.

1. **Thứ tự các tham số có đúng không?**

**Trả lời:** **Thứ tự tham số**: Nếu bạn truyền các tham số vào hàm theo một thứ tự không đúng với thứ tự được khai báo trong hàm, lỗi này sẽ xảy ra. Điều này có thể dẫn đến việc hàm không hoạt động chính xác hoặc trả về kết quả sai.

1. **Thuộc tính của từng tham số thực có tương thích với thuộc tính của tham số**

**hình thức tương ứng của hàm được gọi?**

**Trả lời:** **Tương thích thuộc tính**: Lỗi này xuất phát từ việc các thuộc tính của tham số thực (được truyền vào hàm) không tương thích với thuộc tính của tham số hình thức (được khai báo trong hàm). Điều này có thể dẫn đến việc gây ra lỗi trong quá trình thực thi hoặc kết quả không mong muốn.

1. **Đơn vị đo lường của tham số thực giống với tham số hình thức?**

**Trả lời:**  **Đơn vị đo lường**: Lỗi này xảy ra khi đơn vị đo lường của tham số thực không khớp với đơn vị đo lường của tham số hình thức. Điều này có thể gây ra sai sót trong tính toán hoặc hiển thị kết quả không chính xác.

1. **Tham số read-only có bị thay đổi nội dung bởi hàm không?**

**Trả lời:**  **Tham số read-only:** Tham số được đánh dấu là "read-only" thường không thể bị thay đổi bởi hàm. Nếu hàm cố gắng thay đổi giá trị của tham số này, lỗi sẽ xảy ra.

1. **Định nghĩa biến toàn cục có tương thích giữa các module chức năng không?**

**Trả lời:**  **Biến toàn cục giữa các module:** Lỗi tương thích giữa các module chức năng xuất phát khi bạn cố gắng sử dụng biến toàn cục từ một module trong một module khác mà không có sự khai báo hoặc tương thích. Điều này có thể dẫn đến việc lỗi thực thi hoặc sai sót trong dữ liệu.

**VII - Các lỗi nhập/xuất (Input/Output Errors)**

1. **Lệnh mở/tạo file có đúng chế độ và định dạng truy xuất file?**

**Trả lời:** Lỗi này xảy ra khi bạn sử dụng lệnh để mở hoặc tạo một tệp mà không đúng với chế độ (mode) hoặc định dạng truy cập (access mode) của file. Ví dụ, bạn cố gắng mở một file**\* Lệnh mở/tạo file:** chỉ để đọc nhưng lại sử dụng chế độ ghi.

1. **Kích thước của buffer có đủ chứa dữ liệu đọc vào không?**

**Trả lời: Kích thước buffer đọc**: Khi bạn đọc dữ liệu từ một tệp, bạn thường cần sử dụng một bộ đệm (buffer) để lưu trữ dữ liệu. Nếu kích thước của buffer không đủ lớn để chứa dữ liệu đọc vào, lỗi này có thể xảy ra, dẫn đến việc mất dữ liệu hoặc lỗi đọc.

1. **Có mở file trước khi truy xuất không?**

**Trả lời:**  **Mở file trước khi truy cập:** Trước khi bạn đọc hoặc ghi vào một tệp, bạn cần phải mở tệp đó. Nếu bạn cố gắng thực hiện hoạt động truy cập trên một tệp chưa được mở, lỗi này sẽ xảy ra.

1. **Chuỗi xuất có bị lỗi từ vựng và cú pháp không ?**

**Trả lời:**  **Chuỗi xuất bị lỗi từ vựng và cú pháp:** Khi bạn xuất dữ liệu dạng chuỗi, đôi khi chuỗi này có thể chứa lỗi từ vựng hoặc cú pháp. Điều này có thể xảy ra nếu bạn không đặt các ký tự đặc biệt trong chuỗi vào dấu ngoặc kép, hoặc nếu bạn không xử lý đúng các ký tự thoát (escape characters) như \n cho dòng mới.

**Bài tập:**

**Trả lời:** Lỗi: Trong phần điều kiện của phương thức isLeap() thiếu dấu đóng ngoặc “)” của phần điều kiện if, cần đóng ngoặc “)” sau 0)) để hoàn thành phần điều kiện.