## Code gốc (gây lỗi runtime)

#include <stdio.h>  
int main() {  
 int arr[3] = {1, 2, 3};  
 printf("%d\n", arr[3]); // Lỗi truy cập ngoài mảng  
 return 0;  
}

## Thông báo lỗi khi chạy

Segmentation fault (core dumped)  
hoặc đôi khi in ra giá trị ngẫu nhiên (vì truy cập vùng nhớ không thuộc mảng).

## Prompt đã sử dụng

Hãy phân tích nguyên nhân lỗi từ stack trace, giải thích bằng ngôn ngữ đơn giản và đưa ra bản sửa.

## Phân tích nguyên nhân lỗi (phản hồi từ AI)

Nguyên nhân: Mảng `arr` có 3 phần tử, chỉ số hợp lệ là 0, 1 và 2. Tuy nhiên, chương trình truy cập `arr[3]`, vượt quá phạm vi cấp phát của mảng. Khi đó, CPU cố đọc dữ liệu từ vùng nhớ không thuộc mảng, gây lỗi runtime.  
  
Giải thích đơn giản: Giống như bạn có 3 ngăn tủ được đánh số 0, 1, 2. Nếu bạn cố mở ngăn thứ 4 (`arr[3]`), hệ thống sẽ báo lỗi vì ngăn đó không tồn tại.  
  
Sơ đồ minh họa bộ nhớ:  
 +-----------+-----------+-----------+  
 | arr[0]=1 | arr[1]=2 | arr[2]=3 |  
 +-----------+-----------+-----------+  
 ↑  
 arr[3] ❌ (ngoài phạm vi)  
  
Hệ điều hành sẽ chặn việc truy cập này → lỗi 'Segmentation fault (core dumped)'.  
  
Stack trace (ví dụ khi chạy bằng gdb):  
Program received signal SIGSEGV, Segmentation fault.  
0x000055555555515e in main () at main.c:4

## Code đã sửa (đảm bảo truy cập hợp lệ)

#include <stdio.h>  
int main() {  
 int arr[3] = {1, 2, 3};  
 for (int i = 0; i < 3; i++) { // Duyệt đúng phạm vi  
 printf("%d\n", arr[i]);  
 }  
 return 0;  
}

## Giải thích bản sửa

- Thay vì truy cập trực tiếp `arr[3]`, bản sửa dùng vòng lặp `for` để duyệt đúng phạm vi `[0, 2]`.  
- Như vậy, mọi phần tử đều hợp lệ, chương trình không còn đọc ra vùng nhớ không thuộc mảng.  
- Kết quả in ra: 1, 2, 3.  
  
Nếu muốn in phần tử cuối cùng, có thể dùng: `printf("%d", arr[2]);`.

## Tổng kết

Lỗi 'Segmentation fault' trong trường hợp này xuất phát từ truy cập mảng ngoài phạm vi. Prompt yêu cầu 'phân tích từ stack trace và giải thích bằng ngôn ngữ đơn giản' giúp AI không chỉ chỉ ra dòng lỗi mà còn minh họa bằng sơ đồ và ví dụ dễ hiểu, hỗ trợ người học nắm rõ nguyên nhân gốc thay vì chỉ sửa nhanh.