

Quản lý phạm vi các biến trong mã nguồn C

Để quản lý phạm vi các biến/hằng/kiểu dữ liệu trong mã nguồn C ta sẽ dùng cấu trúc dữ liệu cây tổng quát.

Ví dụ minh họa quản lý biến đoạn code Python sau

```
x = 10      # Global scope

def foo():  # Scope foo
    y = 20  # Local in foo
    if x > 5: # Scope if
        z = 30 # Local in if
        print(y, z)

foo()
```

Ta có cây phạm vi tương ứng là

```
[Global]
|
x = 10 — [foo()]
|
y = 20 — [if x > 5]
|
z = 30
```

Ứng dụng cụ thể

a. Kiểm tra tính hợp lệ của biến (Variable Resolution)

- Khi gặp một biến (ví dụ y trong print(y, z)), trình biên dịch/phân tích sẽ:
 - Tìm kiếm biến y trong phạm vi hiện tại (nút if).
 - Nếu không thấy, đi lên cha (foo), tìm thấy y = 20.
 - Nếu vẫn không thấy, tiếp tục lên đến Global.

b. Phát hiện lỗi phạm vi

- Biến không tồn tại:** Nếu không tìm thấy biến ở bất kỳ phạm vi cha nào → báo lỗi Undefined variable.
- Shadowing:** Nếu một biến ở phạm vi con trùng tên với phạm vi cha → cảnh báo đè biến.

Hãy viết chương trình bằng code C/C++ để quản lý các biến/hằng/kiểu dữ liệu dùng cây tổng quát.

Yêu cầu:

- Chương trình sẽ phân tích 1 file mã nguồn C/C++ đơn giản (chỉ chứa 1 file).
- Xây dựng các cây tổng quát quản lý biến/hằng/kiểu dữ liệu.
- Dựa vào cây tổng quát đã xây dựng, hãy tìm và in ra các lỗi/cảnh báo về cách sử dụng biến/hằng/kiểu dữ liệu trong file đó.