Phân biệt các loại biến Local, Global, Extern, Static và Const

Mục lục

[Biến local 1](#_Toc32618066)

[Biến Global và Extern 1](#_Toc32618067)

[Biến Static 4](#_Toc32618068)

[Biến const 4](#_Toc32618069)

# Biến local

* Xuất hiện trong một phạm vi cụ thể.
* Chỉ tồn tại trong hàm mà biến được khai báo.
* Đôi khi biến local được gọi là biến tự động (auto) vì biến được tự động sinh ra khi hàm thực hiện, và tự động biến mât khi kết thúc hàm.

#include <conio.h>

#include <stdio.h>

int i = 0; // Bien Global

void Display();

int main()

{

    Display();

    printf("\ni = %d", i);

    getch();

}

void Display()

{

    for(i = 0; i < 3; i++)

        printf("\ni = %d", i);

}

# Biến Global và Extern

* Biến global là biến được khai báo bên ngoài tất cả các hàm và có giá trị với tất cả các hàm trong chương trình. Tức là các hàm trong chương trình có thể sử dụng biến global để tính toán.
* Biến global tồn tại đến khi nào chương trình kết thúc.
* Có thể định nghĩa 1 biến global trong 1 file (.c/.cpp/.h) và truy cập biến này từ 1 file (.c/.cpp/.h) khác. Để làm điều này, biến phải được khai báo ở cả 2 file và từ khóa **extern**được thêm trong lần khai báo thứ 2.

1. Ví dụ về biến Global:

#include <conio.h>

#include <stdio.h>

int i = 0; // Bien Global

void Display();

int main()

{

    Display();

    printf("\ni = %d", i);

    getch();

}

void Display()

{

    for (i = 0; i < 3; i++)

        printf("\ni = %d", i);

}

1. Ví dụ biến Extern

* File header1.h

#ifndef \_HEADER\_H\_

#define \_HEADER\_H\_

extern int X;

void display\_x();

#endif

* File file1.cpp

#include "header1.h"

#include "conio.h"

int X; // khai bao bien global

void main()

{

    X = 5;

    display\_x();

    getch();

}

* File file2.cpp

#include "stdio.h"

#include "header1.h"

void display\_x()

{

    printf("X = %d", X);

}

# Biến Static

* Phải khai báo với từ khoá **static**
* Có thể là *global* hoặc *local*
* Biến **local static** là biến có thể duy trì giá trị từ lần gọi hàm thứ nhất đến các lần gọi hàm tiếp theo. Biến **local static** tồn tại đến khi chương trình kết thúc.
* Khi tạo 1 biến **local static** trong hàm, chúng ta nên khởi tạo giá trị cho chúng. Nếu không giá trị biến được gán mặc định bằng 0.
* Biến **global static** là biến *global* mà chỉ có thể truy cập từ file (.c/.cpp) mà biến đó được định nghĩa.

# Biến const

* Trong ngôn ngữ C, tiền xử lý #define được sử dụng để tạo biến với giá trị là hằng số.
* Trong ngôn ngữ C++, xuất hiện một số vấn đề: khi sử dụng #define, tiền xử lí sẽ nhảy thẳng vào source code và thay thế biến bằng giá trị đã định nghĩa. Vì biến #define chỉ tồn tại bên trong file mà nó được định nghĩa, có thể xảy ra trường hợp định nghĩa tên biến giống nhưng khác về giá trị.
* Định nghĩa biến hằng trong C++, chúng ta nên sử dụng từ khóa **const** đi kèm.
* Khi sử dụng từ khóa **const**, phải khởi tạo giá trị ban đầu cho biến.