* **SetUnhanledExceptionFilter()**

Load file vào Ida 32bit, thực hiện debug thì gặp exception Integer divide by zero.



Exception này gây ra tại “div eax”



Đây là ý đồ của chương trình vì ở phía trên chương trình có gọi API SetUnhandledExceotionFilter để đăng kí handle exception với handler là TopLevelExceptionFilter.

 và flow của chương trình sẽ được tiếp tục khi exception này được handle (khi chương trình handle exception thì hàm này sẽ được call, và nếu có debuger thì exception này được handle bởi debuger chứ không phải chương trình, do vậy hàm kia không được call và chương trình không tiếp tục).

Để bypass trick này, ta patch program jump thẳng tới TopLevelExceptionFilter.

2.

F5 để decompiler update lại chương trình.



Sau đó chương trình tạo 1 dialog box với procedure là DialogFunc.



Mặc dù IDA detect sai các tham số của DialogFunc:



tuy nhiên dựa vào calling convention, ta có thể biết được offset +0xC là MessageId của winapi chứ không phải là HWND, tạm thời sửa tên cho dễ nhìn.

Sau khi sửa xong thì đã dễ nhìn hơn rất nhiều, có thể thấy DialogFunc handle 3 window message là WM\_CLOSE, WM\_INITDIALOG và WM\_COMMAND





Khi handle WM\_INITDIALOG, chương trình gọi hàm checkTampered để load 782bytes từ địa chỉ 4011AC để kiểm tra xem có bị breakpoint hay patch không (nếu đặt breakpoint thì debugger sẽ patch 1 byte thành int3(CC) để raise exception tại đó). Nếu phát hiện vùng nhớ bị thay đổi thì sẽ thoát.



