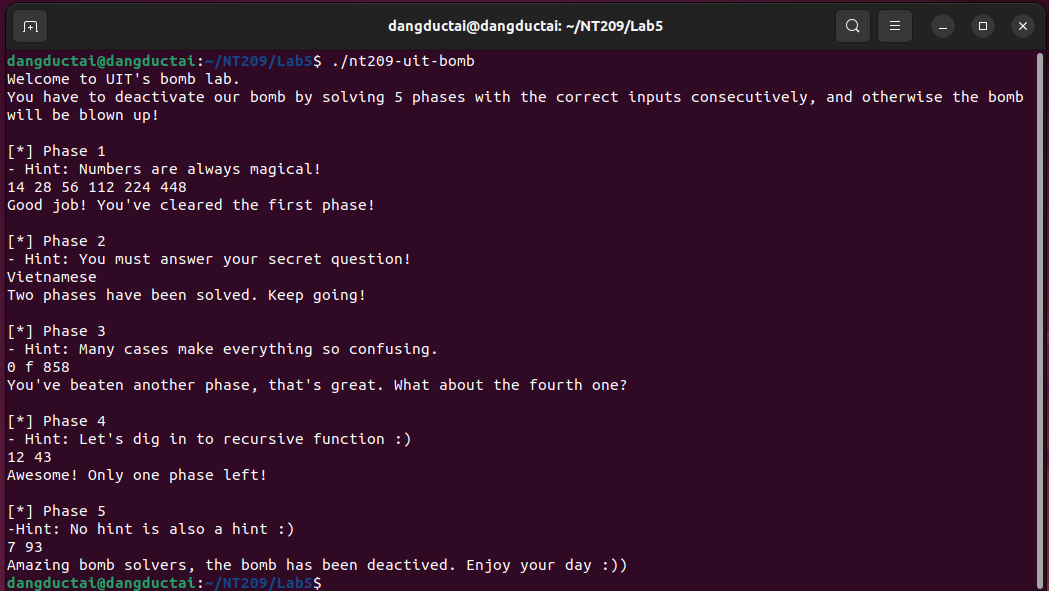
**LẬP TRÌNH HỆ THỐNG  
BÁO CÁO LAB 5**

**KỸ THUẬT DỊCH NGƯỢC (TIẾP THEO)**

|  |  |  |
| --- | --- | --- |
| **Họ và tên** | **MSSV** | **Lớp** |
| Lại Quan Thiên | 22521385 | NT209.O21.ANTT.1  Nhóm 6 |
| Đặng Đức Tài | 22521270 |

# Minh chứng hoàn thành đúng 5 phases:

****

Hình 1: Minh chứng đáp án 5 phases

# Phase 1:

**A white background with green and blue text

Description automatically generated**

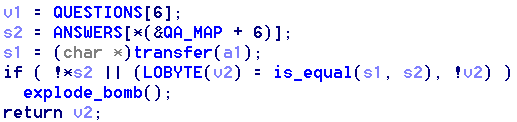
Hình 2: Mã giả của phase 1

**Giải thích:**

* Từ 2 dòng đầu, ta suy ra được rằng, input phải là một dãy gồm 6 số.
* Dòng ***if ( v2[0] != 14)***cho ta biết số đầu tiên là 14.
* Trong vòng for, ta suy ra được rằng ***v2[i] = 2 \* v2[i – 1]***. Tức là số sau bằng 2 lần số trước. Biết số đầu tiên là 14, ta suy ra được kết quả cần tìm là:

***14 28 56 112 224 448***

# Phase 2:



Hình 3: Mã giả của phase 2

**Giải thích:**

* Việc ta cần làm ở phase 2 là trả lời câu hỏi thứ 6 trong mảng QUESTIONS.

A screenshot of a computer

Description automatically generated

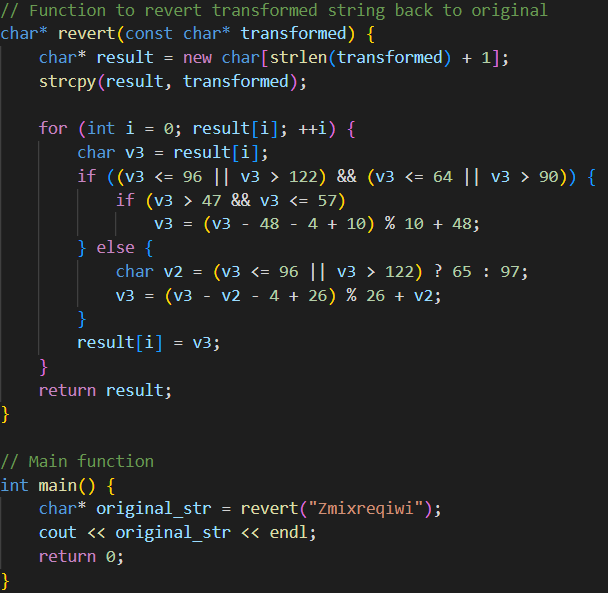
Hình 4: Mảng QUESTIONS

* Câu hỏi chúng ta phải trả lời là: *What is the main language used in this?*
* Chúng ta có thể dễ dàng đưa ra câu trả lời là ***Vietnamese*** nhưng giờ hãy thử tìm ra câu trả lời bằng cách dịch ngược theo đúng yêu cầu bài lab.
* Câu trả lời sẽ là phần tử thứ 6 trong mảng ANSWERS.

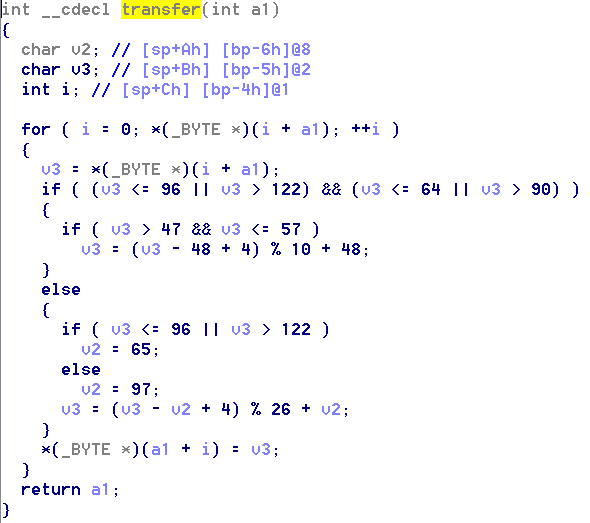
A screenshot of a computer

Description automatically generated

Hình 5: Mảng ANSWERS

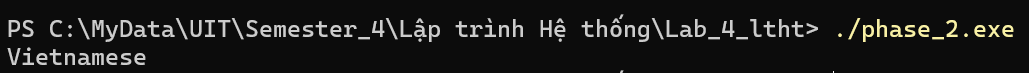
* Câu trả lời đã thị thay đổi bởi hàm ***transfer*** là ***Zmixreqiwi*** nhưng ta cần dịch ngược để có câu trả lời đúng.
* Dựa theo hàm ***transfer***, viết một hàm ***revert*** để tìm ra câu trả lời ban đầu:

Hình 7: Hàm revert



Hình 6: Hàm transfer

* Sau khi revert, ta được kết quả là ***Vietnamese:***



Hình 8: Kết quả phase 2 sau khi revert

# Phase 3:

A screenshot of a computer code

Description automatically generatedA screenshot of a computer code

Description automatically generated

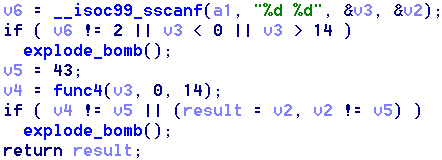
Hình 9: Mã giả của phase 3

**Giải thích:**

* Có 8 cases nên sẽ có 8 đáp án khác nhau.
* Nhìn hai dòng v5 ta dễ dàng nhận ra rằng input là 2 số (%d) và 1 kí tự (%c).
* Số thứ nhất (v4) sẽ là số của từng case. Ví dụ: case 0 thì v4 = 0, case 1 thì v4 = 1...
* Chú ý dòng ***if ( v6 != v2 )***, nếu v6 khác v2 thì bom sẽ nổ, vậy nên v6 phải = v2. Đầu vào của v2 là %c, tức là 1 ký tự nên v6 là mã ASCII của v2. Dựa vào bảng mã ASCII dễ dàng tra ra được v2.
* Giả sử ở case 0, ***if (v3 != 858)***, nếu v3 khác 858 thì bom nổ nên v3 phải bằng 858. Vậy ở mỗi case, v3 phải bằng số đang được so sánh với nó trong trong hàm if.
* Từ các điều kiện trên, ta suy ra kết quả của các case lần lượt là:

|  |  |  |  |
| --- | --- | --- | --- |
| Case 0: ***0 f 858*** | Case 2: ***2 k 913*** | Case 4: ***4 x 465*** | Case 6: ***6 c 186*** |
| Case 1: ***1 y 57*** | Case 3: ***3 u 973*** | Case 5: ***5 n 797*** | Case 7: ***7 a 598*** |

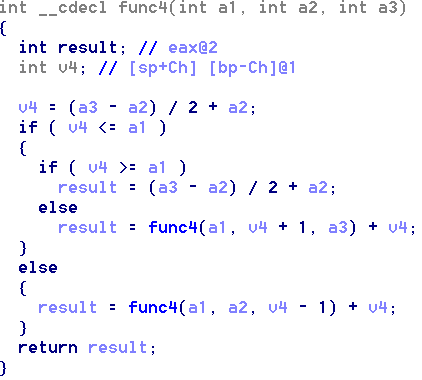
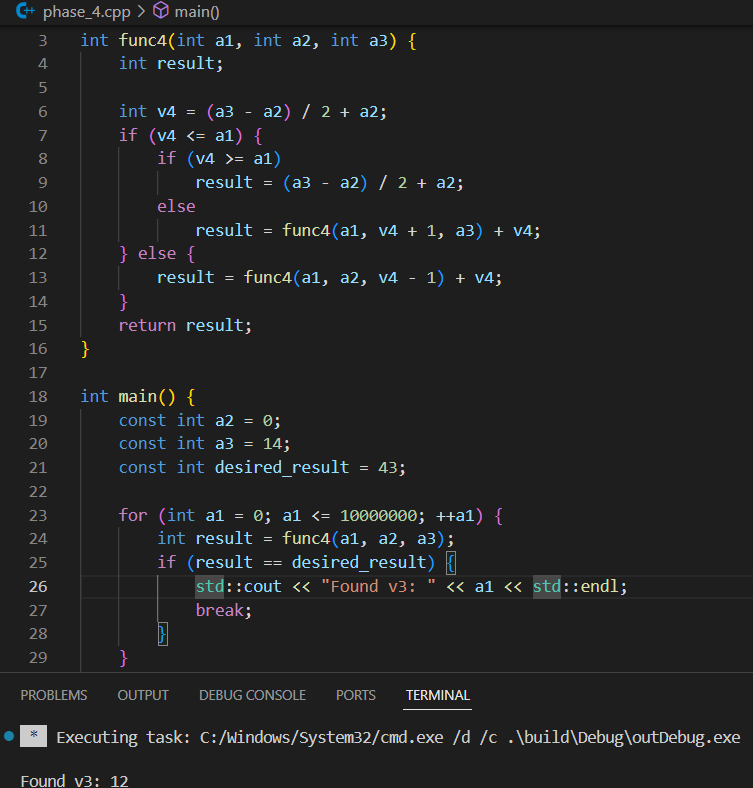
# Phase 4:



Hình 10: Mã giả của phase 4

**Giải thích:**

* Từ dòng đầu tiên, suy ra được input là hai số v3 và v2.
* Dòng thứ hai, nếu v3 < 0 hoặc v3 > 14 bom sẽ nổ nên 0 < v3 < 14.
* Từ dòng if, nếu v4 khác v5 và v2 khác v5 bom sẽ nổ nên v4 phải bằng v5 và v2 phải bằng v5.
* Ta có v5 = 43, suy ra v2 = 43 và v4 =43.
* Ta thấy: ***v4 = func4(v3, 0, 14)****.* Ta đã tìm được v4 = 43. Hàm func4 ta truyền ba tham số là v3, 0, 14. Vậy ***43 = func4(v3, 0, 14).*** Tìm đến func4, viết chương trình bằng C++ để brute force v3 từ hàm func4:



Hình 11: Hàm func4

Hình 12: Brute force phase 4

* Vậy kết quả cần tìm là: ***12 43***

# Phase 5:

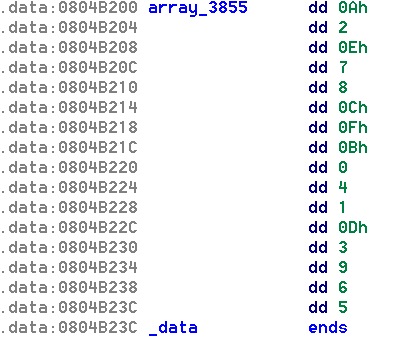
A screenshot of a computer code

Description automatically generated

Hình 13: Mã giả của phase 5

**Giải thích:**

* Từ hai dòng đầu tiên, suy ra input là 2 số.
* Trong vòng lặp while, v7 đóng vai trò là biến index. Ở dòng if, ta suy ra được khi kết thúc vòng lặp v7 = 12, tức là while đã lặp 12 lần và v6 = v2.
* Tìm đến mảng array\_3855 ta thấy:

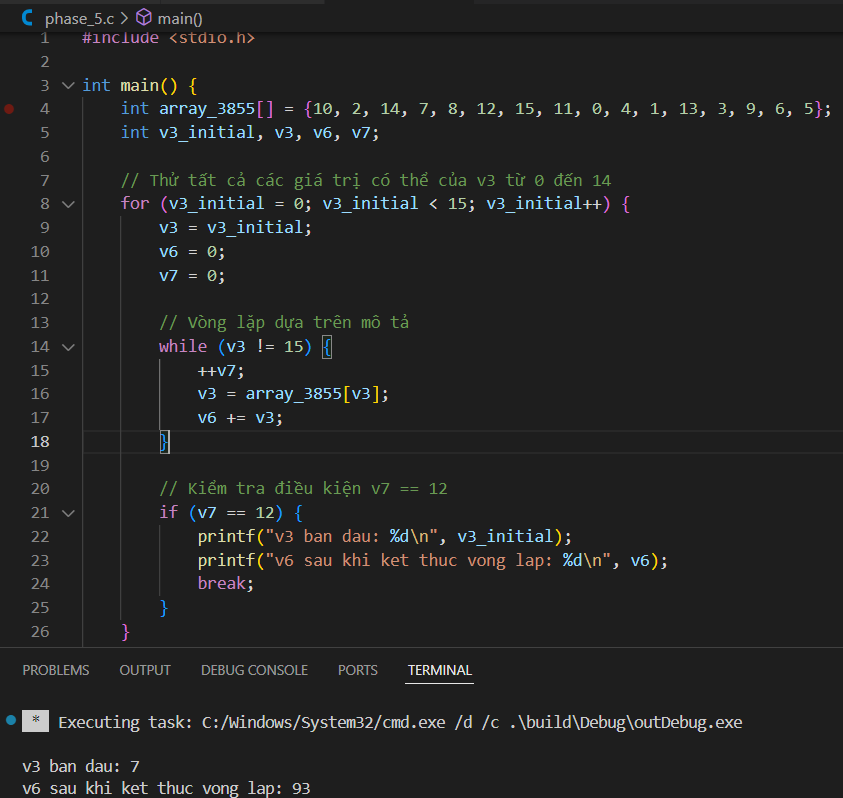


Hình 14: Mảng array\_3855

* Các giá trị có đuôi ***“h”*** là giá trị trong hệ HEX. Chuyển toàn bộ về DEC ta có:

***int array\_3855[] = {10, 2, 14, 7, 8, 12, 15, 11, 0, 4, 1, 13, 3, 9, 6, 5};***

* Vòng lặp while chỉ chạy khi v3 != 15 và kết thúc khi v7 = 12. Do v3 &= 0xFu nên v3 sẽ có giá trị từ 0 đến 15. Giá trị ta cần tìm sẽ là v3 ban đầu và v6 sau khi kết thúc vòng lặp. Viết chương trình bằng C để brute force tìm ra v3 và v6:



Hình 15: Brute force phase 5

* Vậy kết quả cần tìm là: ***7 93***