BÁO CÁO THỰC HÀNH

Môn: Lập trình hệ thống

Buổi báo cáo: Lab 05

Lớp: NT209.O22.ANTT.2

THÔNG TIN CHUNG

STT	Họ và Tên	MSSV	Lớp
1	Trần Tuấn Anh	22520080	ATTT2022.1
2	Nguyễn Khắc Hậu	22520410	ATTT2022.1

Báo Cáo Chi Tiết

Pha 1

Định dạng của input là số và số lượng 6.

```
push    ebp
mov     ebp, esp
sub     esp, 38h
lea     eax, [ebp+var_2C]
add     eax, 14h
push    eax
lea     eax, [ebp+var_2C]
add     eax, 10h
push    eax
lea     eax, [ebp+var_2C]
add     eax, 0Ch
push    eax
lea     eax, [ebp+var_2C]
add     eax, 8
push    eax
lea     eax, [ebp+var_2C]
add     eax, 4
push    eax
lea     eax, [ebp+var_2C]
push    eax
push    offset aDDDDDD  ; "%d %d %d %d %d %d"
push    [ebp+arg_0]
call    ___isoc99_sscanf
add     esp, 20h
```

Điều kiện của Input

```
loc_80489D6:                            ; CODE XREF: phase1+44↑j
                mov     [ebp+var_14], 8
                mov     eax, [ebp+var_2C]
                cmp     eax, [ebp+var_14]
                jnz     short loc_80489F2
                mov     eax, [ebp+var_28]
                mov     edx, [ebp+var_14]
                add     edx, 1
                cmp     eax, edx
                jz      short loc_80489F7

loc_80489F2:                            ; CODE XREF: phase1+58↑j
                call    explode_bomb
```

Ta thấy trong hình câu lệnh cmp input1, 8 nếu không bằng thì gọi hàm explode_bomb → input1 phải bằng 8

Và câu lệnh cmp input2, 9 nếu không bằng thì gọi hàm explode_bomb
→ input2 phải bằng 9

```
loc_80489F7:                            ; CODE XREF: phase1+65↑j
                mov     [ebp+var_C], 2
                jmp     short loc_8048A2A
; ---------------------------------------------------------------------------

loc_8048A00:                            ; CODE XREF: phase1+A3↓j
                mov     eax, [ebp+var_C]
                mov     eax, [ebp+eax*4+var_2C]
                mov     edx, [ebp+var_C]
                sub     edx, 2
                mov     ecx, [ebp+edx*4+var_2C]
                mov     edx, [ebp+var_C]
                sub     edx, 1
                mov     edx, [ebp+edx*4+var_2C]
                add     edx, ecx
                cmp     eax, edx
                jz      short loc_8048A26
                call    explode_bomb
; ---------------------------------------------------------------------------

loc_8048A26:                            ; CODE XREF: phase1+94↑j
                add     [ebp+var_C], 1

loc_8048A2A:                            ; CODE XREF: phase1+73↑j
                cmp     [ebp+var_C], 5
                jle     short loc_8048A00
                nop
                leave
                retn
phase1          endp
```

Trong hình có xuất hiện vòng lặp với giá trị khởi tạo là 2 và điều kiện lặp là <= 5

Ở lần lặp đầu tiên so sánh input3 với input1 + input2

→input3 = input1+input2

Ở lần lặp 2 so sánh input4 với input2 + input3

→input4 = input2+input3

→input sau sẽ bằng tổng 2 input trước

Kết luận input: 8 9 17 26 43 69

```
┌──(kali㉿zasure69)-[~/LTHT/Lab5]
└─$ ./nt209-uit-bomb
Welcome to UIT's bomb lab.
You have to deactivate our bomb by solving 5 phases with the correct inputs consecutively, and otherwise the bomb will be blo
wn up!

[*] Phase 1
- Hint: Numbers are always magical!
8 9 17 26 43 69
Good job! You've cleared the first phase!
```

Pha 2:

Định dạng input là tổ hợp ký tự, số lượng là 1

```
puts("\n[*] Phase 2\n- Hint: You must answer your secret question!");
v4 = read_line();
```

Câu hỏi là câu hỏi thứ QUESTION[0]

```
mov     [ebp+var_C], 0
mov     eax, [ebp+var_C]
mov     eax, QUESTIONS[eax*4]
```

```
public QUESTIONS
QUESTIONS       dd offset aWhatIsTheClass ; DATA XREF: phase2+10↑r
                                          ; "What is the class code of this course?"
                dd offset aWhatIsTheFullE ; "What is the full English name of our un"...
                dd offset aCompleteTheDom ; "Complete the domain of our faculty: htt"...
                dd offset aWhatIsYourMajo ; "What is your major in English? (Capital"...
                dd offset aWhatIsTheEmail ; "What is the email address of your pract"...
                dd offset aWhatIsThePhone ; "What is the phone number of our univers"...
                dd offset aWhatIsTheMainL ; "What is the main language used in this "...
                dd offset aWhatIsTheWeekd ; "What is the weekday that this class tak"...
                dd offset aEnterTheCurren ; "Enter the current date using the format"...
                dd offset aIAmAnOddNumber ; "I am an odd number. Take away one lette"...
                dd offset aWhatIsTheNameO ; "What is the name of the analyzed execut"...
                dd offset aWhatIsYourNati ; "What is your nationality?"
                dd offset aWhatIsTheEstab ; "What is the establishment date of UIT ("...
                dd offset aWhichCityHas37 ; "Which city has 3/7 of a chicken and 2/3"...
                dd offset aWhatIsTheVietn ; "What is the Vietnamese name (without ac"...
                dd offset aMyVehicleRegis ; "My vehicle registration plate starts wi"...
                dd offset aWhatIsTheLarge ; "What is the largest country in the worl"...
                dd offset aWhatWordIsSpel ; "What word is spelled incorrectly in eve"...
                dd offset aIAmOnTheWall_T ; "I am on the wall. Thanks to me, you can"...
                dd offset aWhatIsTheLonge ; "What is the longest wall in the world? "...
```

Điều kiện Input :

```
                mov     eax, QA_MAP[eax*4]
                mov     [ebp+var_14], eax
                mov     eax, [ebp+var_14]
                mov     eax, ANSWERS[eax*4]
                mov     [ebp+s2], eax
                push    [ebp+arg_0]
                call    transfer
                add     esp, 4
                mov     [ebp+s1], eax
                mov     eax, [ebp+s2]
                movzx   eax, byte ptr [eax]
                test    al, al
                jz      short loc_8048A94
                sub     esp, 8
                push    [ebp+s2]        ; s2
                push    [ebp+s1]        ; s1
                call    is_equal
                add     esp, 10h
                test    eax, eax
                jnz     short loc_8048A99

loc_8048A94:                            ; CODE XREF: phase2+4A↑j
                call    explode_bomb
; ---------------------------------------------------------------------------

                public ANSWERS
ANSWERS         dd offset aSy754_t76_fsys ; DATA XREF: phase2+2A↑r
                                        ; "SY754.T76.FSYS.6"
                dd offset aZsnajwxnydTkNs ; "Zsnajwxnyd Tk Nsktwrfynts Yjhmstqtld"
                dd offset aSh_zny_jiz_as ; "sh.zny.jiz.as"
                dd offset aNsktwrfyntsXjh ; "Nsktwrfynts Xjhzwnyd"
                dd offset aMnjsiyy@zny_ji ; "mnjsiyy@zny.jiz.as"
                dd offset a57382707557  ; "57382707557"
                dd offset aAnjysfrjxj   ; "Anjysfrjxj"
                dd offset aYzjxifd      ; "Yzjxifd"
                dd offset a507579       ; "50/7579"
                dd offset aXjajs        ; "Xjajs"
                dd offset aSy754ZnyGtrg ; "sy754-zny-gtrg"
                dd offset aAnjysfrjxj   ; "Anjysfrjxj"
                dd offset a53517551     ; "53/51/7551"
                dd offset aHmnhflt      ; "Hmnhflt"
                dd offset aAfsRnjzVzthYzL ; "Afs Rnjz Vzth Yz Lnfr"
                dd offset aGnsmIztsl    ; "Gnsm Iztsl"
                dd offset aWzxxnf       ; "Wzxxnf"
                dd offset aNshtwwjhyqd  ; "Nshtwwjhyqd"
                dd offset aBnsitb       ; "Bnsitb"
                dd offset aYmjLwjfyBfqqTk ; "Ymj Lwjfy Bfqq Tk Hmnsf"
```

Input nhập vào sau khi xử lý bằng hàm transfer sẽ trở thành chuỗi ANSWER[0] = SY754.T76.FSYS.6

Mã lớp là NT209.XXX.XXXX.X

Ta thấy kí tự mã lớp sẽ nhỏ hơn đáp án 5 đơn vị

→ các ký tự của input = ký tự của chuỗi đáp án - 5

→input = NT209.O21.ANTN.1

```
[*] Phase 2
- Hint: You must answer your secret question!
NT209.O21.ANTN.1
Two phases have been solved. Keep going!
```

Pha 3 :

Có 3 input với định dạng input1 và input3 là số, input2 là ký tự

```
lea     eax, [ebp+var_18]
push    eax
lea     eax, [ebp+var_19]
push    eax
lea     eax, [ebp+var_14]
push    eax
push    offset aDCD     ; "%d %c %d"
push    [ebp+arg_0]
call    ___isoc99_sscanf
```

Điều kiện input :

Switch case input1 với các case từ 0 – 7

```
loc_8048AD6:                            ; CODE XREF: phase3+33↑j
                mov     eax, [ebp+var_14]
                cmp     eax, 7          ; switch 8 cases
                ja      loc_8048BA3     ; jumptable 08048AE9 default case
                mov     eax, ds:off_8049860[eax*4]
                jmp     eax             ; switch jump

loc_8048BA3:                            ; CODE XREF: phase3+40↑j
                mov     [ebp+var_9], 61h ; jumptable 08048AE9 default case
                call    explode_bomb
```

Trường hợp case 0 (input1 = 0):

```
loc_8048AEB:                            ; CODE XREF: phase3+4D↑j
                                        ; DATA XREF: .rodata:off_8049860↓o
                mov     [ebp+var_9], 76h ; jumptable 08048AE9 case 0
                mov     eax, [ebp+var_18]
                cmp     eax, 0A4h
                jz      loc_8048BAE
                call    explode_bomb
; ---------------------------------------------------------------------------
                jmp     loc_8048BAE
; ---------------------------------------------------------------------------

loc_8048BAE:                            ; CODE XREF: phase3+5B↑j
                                        ; phase3+66↑j
                nop
                jmp     short loc_8048BC4
; ---------------------------------------------------------------------------

loc_8048BC4:                            ; CODE XREF: phase3+110↑j
                                        ; phase3+113↑j ...
                movzx   eax, [ebp+var_19]
                cmp     [ebp+var_9], al
                jz      short loc_8048BD2
                call    explode_bomb
```

So sánh input2 với 118 và input3 với 164

→input2 là kí tự có mã ASCII là 118 là ký tự ‘v’

→input3 là 164

→Kết quả input: 0 v 164

```
[*] Phase 3
- Hint: Many cases make everything so confusing.
0 v 164
You've beaten another phase, that's great. What about the fourth one?
```

Tương tự các case khác sẽ có các input :

1 { 862

```
case 1:
  v6 = 123;
  if ( v3 != 862 )
    explode_bomb();
  return result;
```

```
[*] Phase 3
- Hint: Many cases make everything so confusing.
1 { 862
You've beaten another phase, that's great. What about the fourth one?
```

2 l 54

```
case 2:
  v6 = 108;
  if ( v3 != 54 )
    explode_bomb();
  return result;
```

```
[*] Phase 3
- Hint: Many cases make everything so confusing.
2 l 54
You've beaten another phase, that's great. What about the fourth one?
```

3 b 533

```
case 3:
  v6 = 98;
  if ( v3 != 533 )
    explode_bomb();
  return result;
```

```
[*] Phase 3
- Hint: Many cases make everything so confusing.
3 b 533
You've beaten another phase, that's great. What about the fourth one?
```

4 f 176

```
case 4:
  v6 = 102;
  if ( v3 != 176 )
    explode_bomb();
  return result;
```

```
[*] Phase 3
- Hint: Many cases make everything so confusing.
4 f 176
You've beaten another phase, that's great. What about the fourth one?
```

5 h 914

```
case 5:
  v6 = 104;
  if ( v3 != 914 )
    explode_bomb();
  return result;
```

```
[*] Phase 3
- Hint: Many cases make everything so confusing.
5 h 914
You've beaten another phase, that's great. What about the fourth one?
```

6 y 438

```
case 6:
  v6 = 121;
  if ( v3 != 438 )
    explode_bomb();
  return result;
```

```
[*] Phase 3
- Hint: Many cases make everything so confusing.
6 y 438
You've beaten another phase, that's great. What about the fourth one?
```

7 s 826

```
case 7:
  v6 = 115;
  if ( v3 != 826 )
    explode_bomb();
  return result;
```

```
[*] Phase 3
- Hint: Many cases make everything so confusing.
7 s 826
You've beaten another phase, that's great. What about the fourth one?
```

Pha 4 :

Định dạng input là số, số lượng là 2

```
sub     esp, 28h
lea     eax, [ebp+var_1C]
push    eax
lea     eax, [ebp+var_18]
push    eax
push    offset aDD      ; "%d %d"
push    [ebp+arg_0]
call    ___isoc99_sscanf
```

Điều kiện input :

```
mov     [ebp+var_C], eax
cmp     [ebp+var_C], 2
jnz     short loc_8048C7C
mov     eax, [ebp+var_18]
test    eax, eax
js      short loc_8048C7C
mov     eax, [ebp+var_18]
cmp     eax, 0Eh
jle     short loc_8048C81
```

Input1 >= 0 và Input1 <= 14

```
push    0Eh
push    0
push    eax
call    func4
add     esp, 10h
mov     [ebp+var_14], eax
mov     eax, [ebp+var_14]
cmp     eax, [ebp+var_10]
jnz     short loc_8048CAE
mov     eax, [ebp+var_1C]
cmp     eax, [ebp+var_10]
jz      short loc_8048CB3

loc_8048CAE:                            ; CODE XREF
call    explode_bomb
; ---------------------------------------------------
```

So sánh Input2 với 0 → Input2 = 0

So sánh giá trị trả về của hàm func4(input1, 0, 14) = 0

```
int __cdecl func4(int a1, int a2, int a3)
{
  int result; // eax@2
  int v4; // [sp+Ch] [bp-Ch]@1

  v4 = (a3 - a2) / 2 + a2;
  if ( v4 <= a1 )
  {
    if ( v4 >= a1 )
      result = 0;
    else
      result = 2 * func4(a1, v4 + 1, a3) + 1;
  }
  else
  {
    result = 2 * func4(a1, a2, v4 - 1);
  }
  return result;
}
```

Hàm sẽ trả về 0 nếu input1 = hiệu 2 tham số sau / 2 + tham số thứ 2

Input thỏa mãn:

7 0

```
[*] Phase 4
- Hint: Let's dig in to recursive function :)
7 0
Awesome! Only one phase left!
```

3 0

```
[*] Phase 4
- Hint: Let's dig in to recursive function :)
3 0
Awesome! Only one phase left!
```

1 0

```
[*] Phase 4
- Hint: Let's dig in to recursive function :)
1 0
Awesome! Only one phase left!
```

Pha 5:

Định dạng input là số, số lượng là 2

```
lea     eax, [ebp+var_20]
push    eax
lea     eax, [ebp+var_1C]
push    eax
push    offset aDD      ; "%d %d"
push    [ebp+arg_0]
call    ___isoc99_sscanf
add     esp, 10h
```

Điều kiện input

```
loc_8048D01:                            ; CODE
                add     [ebp+var_C], 1
                mov     eax, [ebp+var_1C]
                mov     eax, array_3855[eax*4]
                mov     [ebp+var_1C], eax
                mov     eax, [ebp+var_1C]
                add     [ebp+var_10], eax

loc_8048D18:                            ; CODE
                mov     eax, [ebp+var_1C]
                cmp     eax, 0Fh
                jnz     short loc_8048D01
                cmp     [ebp+var_C], 9
                jnz     short loc_8048D2E
                mov     eax, [ebp+var_20]
                cmp     [ebp+var_10], eax
                jz      short loc_8048D33

loc_8048D2E:                            ; CODE
                call    explode_bomb
```

input1 phải thỏa để chạy vòng lặp với điều kiện lặp là input1 != 15 đủ 9 lần và input2 = tổng các input1 trong các lần lặp

input1 sẽ thay đổi giá trị dựa vào mảng array_3855

```
.data:0804B1A0 ; int array_3855[]
.data:0804B1A0 array_3855      dd 0Ah
.data:0804B1A4                 db    2
.data:0804B1A5                 db    0
.data:0804B1A6                 db    0
.data:0804B1A7                 db    0
.data:0804B1A8                 db  0Eh
.data:0804B1A9                 db    0
```

Kết luận input:

input1 khi lặp lần cuối sẽ là 15 = array_3855[6] →input1 lần lặp trước là 6

```
.data:0804B1B7                 db    0
.data:0804B1B8                 db  0Fh
.data:0804B1B9                 db    0
.data:0804B1BA                 db    0
.data:0804B1BB                 db    0
.data:0804B1BC                 db  0Bh
.data:0804B1BD                 db    0
```

tương tự như vậy sẽ suy ra được input1 = 9 và cộng các input1 trong vòng lặp→input2 = 60

Kết luận input: 9 60

```
[*] Phase 5
-Hint: No hint is also a hint :)
9 60
Amazing bomb solvers, the bomb has been deactived. Enjoy your day :))
```