

BÁO CÁO BÀI TẬP

Môn học: Cơ chế hoạt động của mã độc

Tên chủ đề: Lab 1:

ÔN TẬP NGÔN NGỮ ASSEMBLY & CHÈN MÃ VÀO TẬP TIN PE

GVHD: Nguyễn Hữu Quyền

1. THÔNG TIN CHUNG:

(Liệt kê tất cả các thành viên trong nhóm)

Lớp: NT230.022.ATCL.1

STT	Họ và tên	MSSV	Email
1	Nguyễn Văn Anh Tú	21520156	21520156@gm.uit.edu.vn
2	Phạm Thanh Tâm	21522573	21522573@gm.uit.edu.vn
3	Lâm Hải Đăng	21520682	21520682@gm.uit.edu.vn
4	Nguyễn Đình Kha	21520948	21520948@gm.uit.edu.vn

2. NỘI DUNG THỰC HIỆN:¹

STT	Công việc	Kết quả tự đánh giá
1	Bài 1	100%
2	Bài 2	100%
3	Bài 3	100%
4	Bài 4	100%
5	Bài 5	100%

Phần bên dưới của báo cáo này là tài liệu báo cáo chi tiết của nhóm thực hiện.

¹ Ghi nội dung công việc, các kịch bản trong bài Thực hành

BÁO CÁO CHI TIẾT

Bài thực hành 1: Viết một đoạn chương trình tìm số nhỏ nhất trong 3 số (1 chữ số) a,b,c cho trước.

Full code:

```
1  SYS_EXIT equ 1
2  SYS_READ equ 3
3  SYS_WRITE equ 4
4  STDIN equ 0
5  STDOUT equ 1
6
7  section .text
8      global _start          ;must be declared for using gcc
9
10 _start:                    ;tell linker entry point
11     ; Input number 1
12     mov eax, SYS_WRITE     ; syscall write
13     mov ebx, STDOUT        ; file descriptor (stdout)
14     mov ecx, msg1          ; move message msg1 to ecx
15     mov edx, len1          ; move length of msg1 to edx
16     int 0x80              ; call kernel
17     mov eax, SYS_READ      ; system call read
18     mov ebx, STDIN         ; file descriptor (stdin)
19     mov ecx, num1          ; move num1 to ecx
20     mov edx, 4             ; move size of num1 to edx
21     int 0x80              ; call kernel
22     ; Input num 2 + 3: Code similar num 1
23     mov eax, SYS_WRITE
24     mov ebx, STDOUT
25     mov ecx, msg2
26     mov edx, len2
27     int 0x80
```

```
28     mov eax, SYS_READ
29     mov ebx, STDIN
30     mov ecx, num2
31     mov edx, 4
32     int 0x80
33
34     mov eax, SYS_WRITE
35     mov ebx, STDOUT
36     mov ecx, msg3
37     mov edx, len3
38     int 0x80
39     mov eax, SYS_READ
40     mov ebx, STDIN
41     mov ecx, num3
42     mov edx, 4
43     int 0x80
44
45     ; num1 -> eax, num2 -> ebx; num3 -> ecx
46     ; Find the smallest
47     mov  eax, [num1]
48     sub  eax, '0'
49     mov  ebx, [num2]
50     sub  ebx, '0'
51     mov  ecx, [num3]
52     sub  ecx, '0'
53     ; Compare num1 & num2
54     cmp  eax, ebx
55     jl   check_third_num ; if num1 < num2 => Compare num1 & num3
```

```
56     mov    eax, ebx ; else move num2 -> eax => Compare num2 & num3
57
58     check_third_num: ; Compare numX & num 3
59
60     cmp    eax, ecx
61     jl     _exit      ; if numX < num3 => numX min => exit
62     mov    eax, ecx   ; else num3 min => move num3 -> eax
63
64
65     _exit: ; exit function
66
67     add    eax, '0'   ; add '0' to to convert the result from decimal to ASCII
68     mov    [smallest], eax ; move result to smallest
69     mov    ecx, msg    ; move message msg to ecx
70     mov    edx, len    ; move length of message msg to edx
71     mov    ebx, 1      ;file descriptor (stdout)
72     mov    eax, 4      ;system call number (sys_write)
73     int    0x80        ;call kernel
74
75     mov    ecx, smallest ; move result to ecx for printing
76     mov    edx, 4      ; move size of result to edx
77     mov    ebx, 1      ;file descriptor (stdout)
78     mov    eax, 4      ;system call number (sys_write)
79     int    0x80        ;call kernel
80
81     mov    eax, 1      ; syscall EXIT
82     int    80h         ; call kernal
```

```

84 section .data
85
86 msg1 db "Please enter first digit: ", 0xA,0xD ; define message msg1
87 len1 equ $- msg1 ; define len1 = length of message msg1
88 msg2 db "Please enter a second digit", 0xA,0xD ; define message msg2
89 len2 equ $- msg2 ; define len2 = length of message msg2
90 msg3 db "Please enter a third digit", 0xA,0xD ; define message msg3
91 len3 equ $- msg3 ; define len3 = length of message msg3
92
93
94 msg db "The smallest digit is: ", 0xA,0xD ; define message msg
95 len equ $- msg ; define len = length of message msg
96
97
98 segment .bss
99 num1 resb 4 ; define num1 with size = 4
100 num2 resb 4 ; define num2 type d with size = 4
101 num3 resb 4 ; define num3 type d with size = 4
102 smallest resb 4 ; define smallest type d with size = 4

```

Kết quả:

<pre> 1 SYS_EXIT equ 1 2 SYS_READ equ 3 3 SYS_WRITE equ 4 4 STDIN equ 0 5 STDOUT equ 1 6 7 section .text 8 global _start ;must be declared for using gcc 9 10 _start: ;tell linker entry point 11 ; Input number 1 12 mov eax, SYS_WRITE ; syscall write 13 mov ebx, STDOUT ; file descriptor (stdout) 14 mov ecx, msg1 ; move message msg1 to ecx 15 mov edx, len1 ; move length of msg1 to edx 16 int 0x80 ; call kernel 17 mov eax, SYS_READ ; system call read 18 mov ebx, STDIN ; file descriptor (stdin) 19 mov ecx, num1 ; move num1 to ecx 20 mov edx, 4 ; move size of num1 to edx 21 int 0x80 ; call kernel 22 ; Input num 2 + 3: Code similar num 1 23 mov eax, SYS_WRITE 24 mov ebx, STDOUT 25 mov ecx, msg2 26 mov edx, len2 27 int 0x80 </pre>	<pre> Please enter first digit: 5 Please enter a second digit 7 Please enter a third digit 9 The smallest digit is: 5 .. </pre>
--	---

- Chương trình bắt đầu bằng việc yêu cầu người dùng nhập 3 số bất kì num1, num2, num3 có 1 chữ số, sau đó chuyển chúng thành mã ASCII và so sánh với nhau.
- Sau đó so sánh num1 và num2 nếu số nào nhỏ hơn thì so sánh với num3 để chọn ra số nhỏ nhất và hiển thị ra màn hình. Cụ thể giải thích câu lệnh đã có trong đoạn code

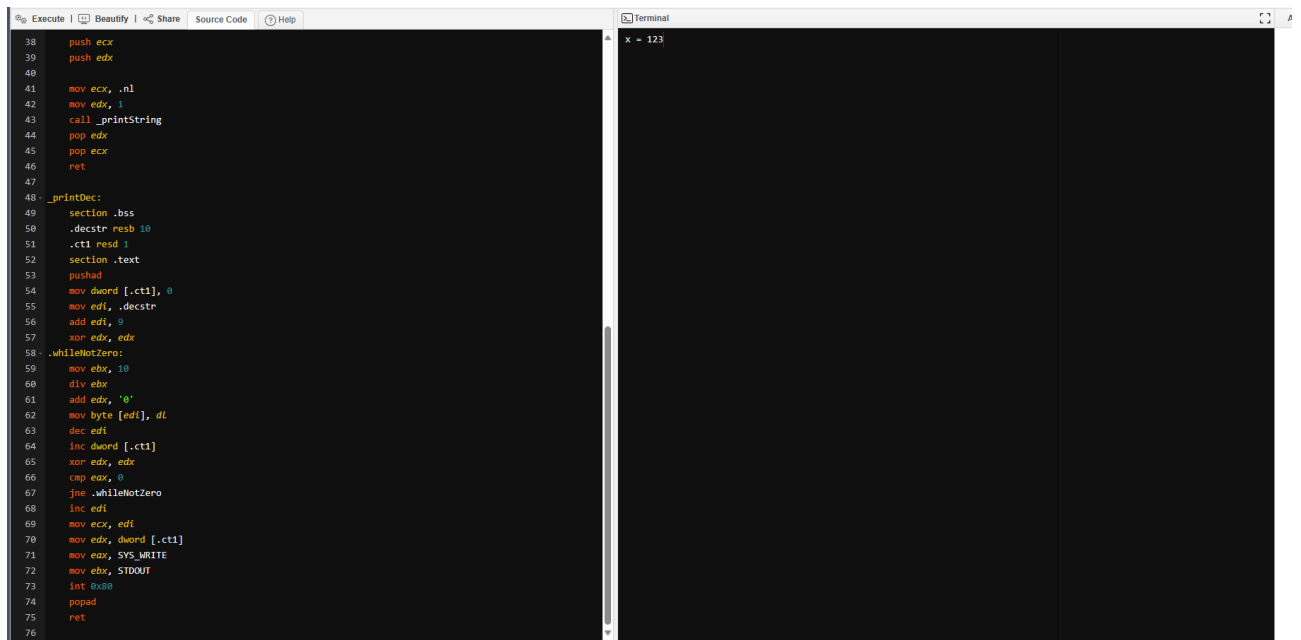
Bài thực hành 2: Viết chương trình chuyển đổi một số (number) 123 thành chuỗi '123'
Sau đó thực hiện in ra màn hình số 123.

Code:

```
Execute | Beautify | Share | Source Code | Help
1  %assign SYS_EXIT 1
2  %assign SYS_WRITE 4
3  %assign STDOUT 1
4
5  section .data
6  x db 123
7  msgX db "x = "
8
9  section .text
10 global _start
11 _start:
12     mov ecx, msgX
13     mov edx, 4
14     call _printString
15     mov eax, 0
16     mov al, byte[x]
17     call _printDec
18     mov ebx, 0
19     mov eax, 1
20     int 0x80
21
22 _printString:
23     push eax
24     push ebx
25
26     mov eax, SYS_WRITE
27     mov ebx, STDOUT
28     int 0x80
29     pop ebx
30     pop eax
31     ret
32
33 println:
34     section .data
35     .nl db 10
36
37     section .text
38     push ecx
39     push edx
```

```
Execute | Beautify | Share | Source Code | Help
38     push ecx
39     push edx
40
41     mov ecx, .nl
42     mov edx, 1
43     call _printString
44     pop edx
45     pop ecx
46     ret
47
48 - _printDec:
49     section .bss
50     .decstr resb 10
51     .ct1 resd 1
52     section .text
53     pushad
54     mov dword [.ct1], 0
55     mov edi, .decstr
56     add edi, 9
57     xor edx, edx
58 - .whileNotZero:
59     mov ebx, 10
60     div ebx
61     add edx, '0'
62     mov byte [edi], dl
63     dec edi
64     inc dword [.ct1]
65     xor edx, edx
66     cmp eax, 0
67     jne .whileNotZero
68     inc edi
69     mov ecx, edi
70     mov edx, dword [.ct1]
71     mov eax, SYS_WRITE
72     mov ebx, STDOUT
73     int 0x80
74     popad
75     ret
76
```

Kết quả chạy:



The screenshot shows a code editor on the left and a terminal on the right. The code editor contains assembly code for a program that prints the decimal string "123". The terminal shows the output "x = 123".

```
38  push ecx
39  push edx
40
41  mov ecx, .nl
42  mov edx, 1
43  call _printf
44  pop edx
45  pop ecx
46  ret
47
48  _printf:
49  section .bss
50  .decstr resb 10
51  .cti resd 1
52  section .text
53  pushad
54  mov dword [.cti], 0
55  mov edi, .decstr
56  add edi, 9
57  xor edx, edx
58  .whileNotZero:
59  mov ebx, 10
60  div ebx
61  add edx, '0'
62  mov byte [edi], dl
63  dec edi
64  inc dword [.cti]
65  xor edx, edx
66  cmp eax, 0
67  jne .whileNotZero
68  inc edi
69  mov ecx, edi
70  mov edx, dword [.cti]
71  mov eax, SYS_WRITE
72  mov ebx, STDOUT
73  int 0x80
74  popad
75  ret
76
```

Terminal output: x = 123

Giải thích đoạn code: Đầu tiên, ta có được số 123 cho sẵn, để có thể chuyển nó thành string, ta cần phải tách các chữ số ra bằng cách tuần tự chia số cho 10, sau đó chuyển các chữ số này sang mã ASCII và ghép lại thành chuỗi "123".

Bài thực hành 3: Cải tiến chương trình yêu cầu 1 sao cho tìm số nhỏ nhất trong 3 số bất kỳ (nhiều hơn 1 chữ số)

- Code và kết quả:


```
Execute | Beautify | Share | Source Code | Help

1  SYS_EXIT equ 1
2  SYS_READ equ 3
3  SYS_WRITE equ 4
4  STDIN equ 0
5  STDOUT equ 1
6
7  section .text
8      global _start
9
10 ~ _start:
11     ;INPUT NUMBER;
12     mov eax, SYS_WRITE
13     mov ebx, STDOUT
14     mov ecx, msg1
15     mov edx, len1
16
17     int 0x80
18     mov eax, SYS_READ
19     mov ebx, STDIN
20     mov ecx, num1
21     mov edx, 4
22
23     int 0x80
24     mov eax, SYS_WRITE
25     mov ebx, STDOUT
26     mov ecx, msg2
27     mov edx, len2
28
29     int 0x80
30     mov eax, SYS_READ
31     mov ebx, STDIN
32     mov ecx, num2
33     mov edx, 4
34
35     int 0x80
36     mov eax, SYS_WRITE
37     mov ebx, STDOUT
38     mov ecx, msg3
39     mov edx, len3
```

```
Execute | Beautify | Share | Source Code | Help

32     mov ecx, num2
33     mov edx, 4
34
35     int 0x80
36     mov eax, SYS_WRITE
37     mov ebx, STDOUT
38     mov ecx, msg3
39     mov edx, len3
40
41     int 0x80
42     mov eax, SYS_READ
43     mov ebx, STDIN
44     mov ecx, num3
45     mov edx, 4
46     int 0x80
47
48     mov eax, [num1]
49     sub eax, '0'
50     mov ebx, [num2]
51     sub ebx, '0'
52     mov ecx, [num3]
53     sub ecx, '0'
54
55     ;COMPARE num1 and num2;
56     cmp eax, ebx
57     jl check_third_num ;if num1 < num2 => check num1 and num3;
58     mov eax, ebx ;else move num2 to eax => check num2 and num3;
59
60     check_third_num: ;check (num1 or num2) and num3;
61
62     cmp eax, ecx
63     jl smallest_num ;
64     mov eax, ecx
65
66     smallest_num:
67     add eax, '0' ; Convert back to ASCII character
68     mov [smallest], eax ;move result to the smallest
69     mov ecx, msg
70     mov edx, len
```

The screenshot shows a code editor on the left and a terminal window on the right. The code editor contains assembly code for finding the smallest of three numbers. The terminal shows the execution of the program with user input and the resulting output.

```

62  cmp eax, ecx
63  jl smallest_num ;
64  mov ecx, ecx
65
66  smallest_num:
67  add eax, '0' ; Convert back to ASCII character
68  mov [smallest], eax ;move result to the smallest
69  mov ecx, msg
70  mov ecx, len
71  mov ebx, STDOUT
72  mov eax, SYS_WRITE
73  int 0x08
74
75  mov ecx, smallest
76  mov ecx, 4
77  mov ebx, STDOUT
78  mov eax, SYS_WRITE
79  int 0x08
80
81  mov eax, SYS_EXIT
82  int 0x08
83
84  section .data
85  msg1 db "Enter the first number: ", 0xA, 0xD
86  len1 equ $-msg1
87  msg2 db "Enter the second number: ", 0xA, 0xD
88  len2 equ $-msg2
89  msg3 db "Enter the third number: ", 0xA, 0xD
90  len3 equ $-msg3
91
92  msg db "The smallest number is: ", 0xA, 0xD
93  len equ $-msg
94
95  segment .bss
96  num1 resb 4
97  num2 resb 4
98  num3 resb 4
99  smallest resb 4
100

```

The terminal output shows the following sequence of events:

```

Enter the first number:
123
Enter the second number:
456
Enter the third number:
789
The smallest number is:
123

```

- **Giải thích đoạn code:** Chương trình bắt đầu bằng việc yêu cầu người dùng nhập ba số nguyên từ bàn phím, mỗi lần nhập một số. Ba số này được lưu vào ba biến num1, num2, num3.

Sau khi nhập ba số, chương trình tiến hành so sánh num1 và num2. Nếu num1 nhỏ hơn num2, chương trình sẽ so sánh num1 với num3 để tìm số nhỏ nhất. Ngược lại, nếu num1 không nhỏ hơn num2, chương trình sẽ so sánh num2 với num3.

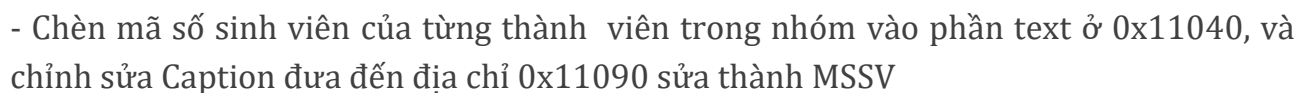
Sau khi xác định được số nhỏ nhất trong hai số đầu tiên, chương trình so sánh số nhỏ nhất với num3. Nếu num3 nhỏ hơn số nhỏ nhất đã tìm được, thì num3 sẽ là số nhỏ nhất cuối cùng.

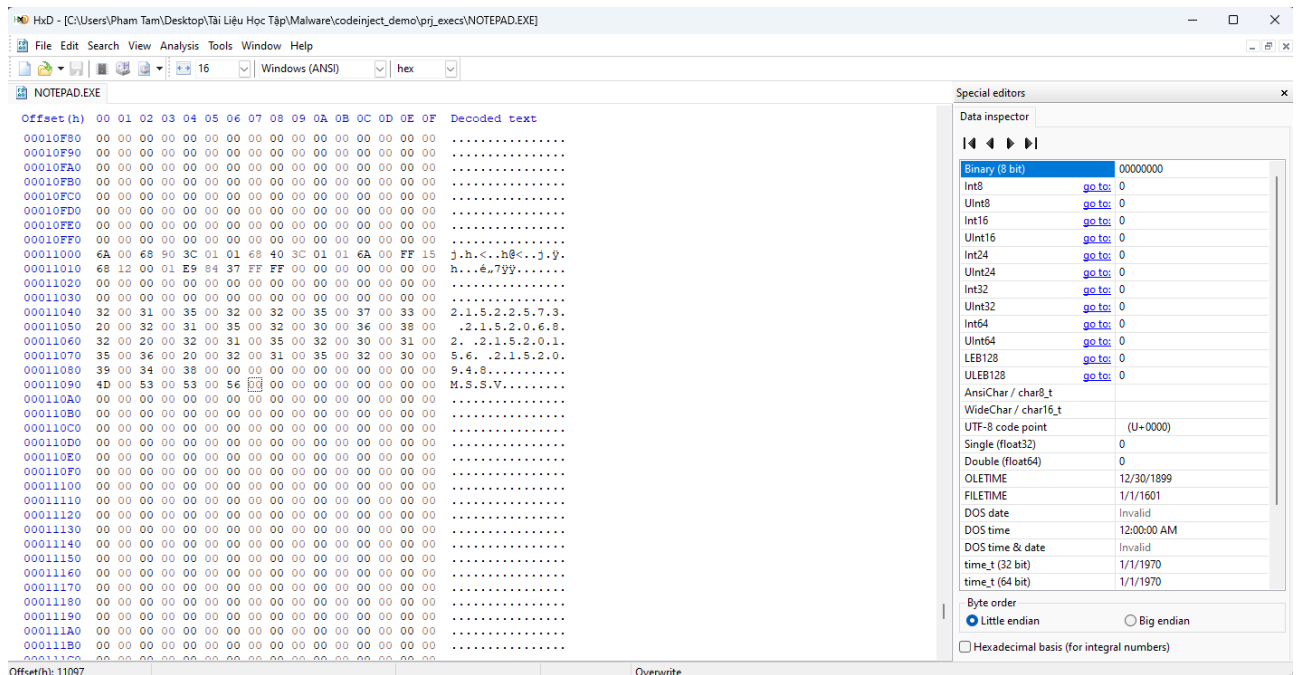
Cuối cùng, chương trình sẽ hiển thị số nhỏ nhất đã tìm được trên màn hình và kết thúc chương trình.

CHÈN MÃ VÀO TẬP TIN PE

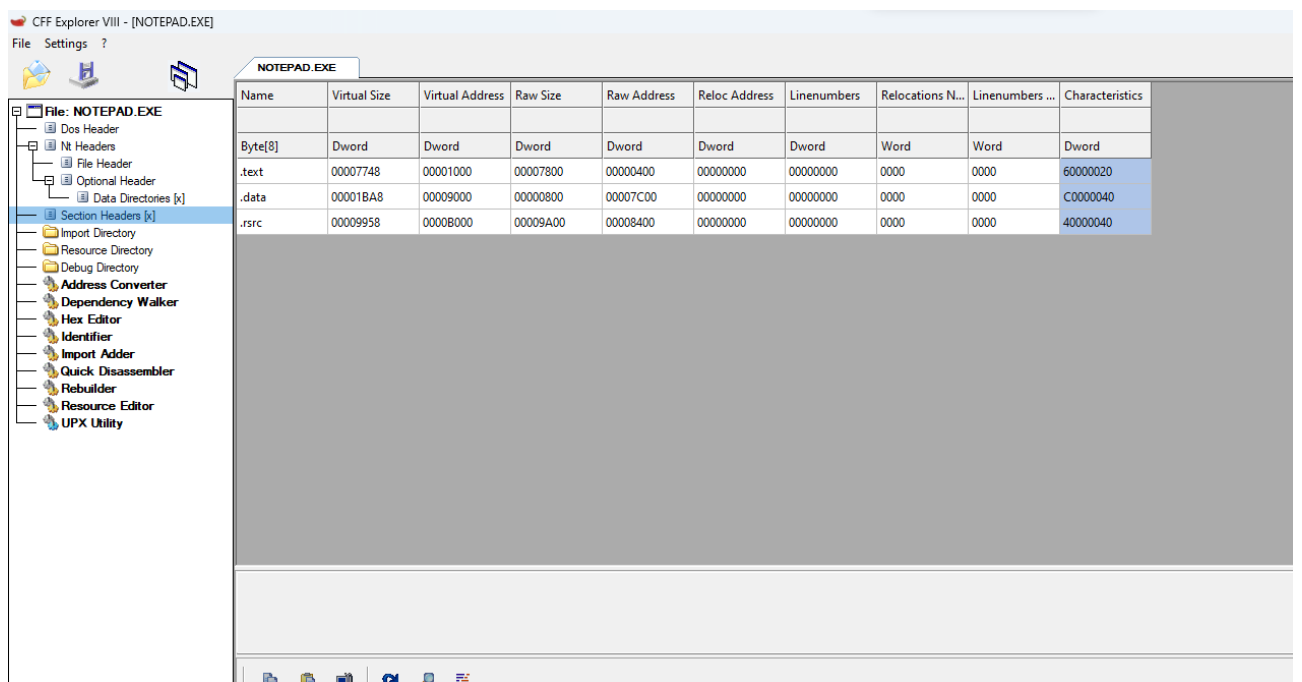
Bài thực hành 4: Thực hiện lại các bước trên thay đổi phần Text là MSSV.

- Thêm vùng nhớ vào tập tin PE





- Lưu lại và mở CFF Explorer của NOTEPAD.EXE lên, chỉnh sửa lại .rsrc Section Header



- Vào Optional Header chỉnh sửa AddressOfEntryPoint thành 0x00013C00, SizeOfImage tăng thêm 0x1000

CFF Explorer VIII - [NOTEPAD.EXE]

File Settings ?

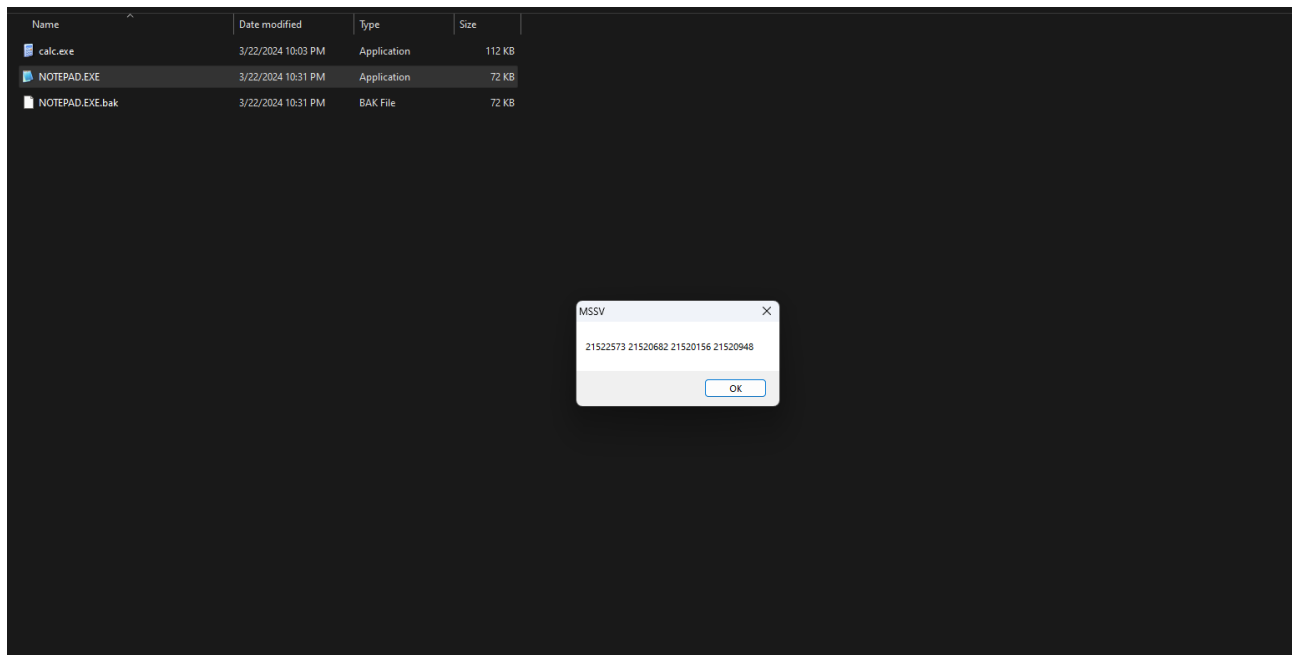
NOTEPAD.EXE

File: NOTEPAD.EXE

- Dos Header
- Nt Headers
 - File Header
 - Optional Header
 - Data Directories [x]
- Section Headers [x]
- Import Directory
- Resource Directory
- Debug Directory
- Address Converter
- Dependency Walker
- Hex Editor
- Identifier
- Import Adder
- Quick Disassembler
- Rebuilder
- Resource Editor
- UPX Utility

Member	Offset	Size	Value	Meaning
Magic	000000F8	Word	010B	PE32
MajorLinkerVersion	000000FA	Byte	07	
MinorLinkerVersion	000000FB	Byte	0A	
SizeOfCode	000000FC	Dword	00007800	
SizeOfInitializedData	00000100	Dword	0000A600	
SizeOfUninitializedData	00000104	Dword	00000000	
AddressOfEntryPoint	00000108	Dword	00013C00	.rsrc
BaseOfCode	0000010C	Dword	00001000	
BaseOfData	00000110	Dword	00009000	
ImageBase	00000114	Dword	01000000	
SectionAlignment	00000118	Dword	00001000	
FileAlignment	0000011C	Dword	00000200	
MajorOperatingSystemVers...	00000120	Word	0005	
MinorOperatingSystemVers...	00000122	Word	0001	
MajorImageVersion	00000124	Word	0005	
MinorImageVersion	00000126	Word	0001	
MajorSubsystemVersion	00000128	Word	0004	
MinorSubsystemVersion	0000012A	Word	0000	
Win32VersionValue	0000012C	Dword	00000000	
SizeOfImage	00000130	Dword	00015000	
SizeOfHeaders	00000134	Dword	00000400	
Checksum	00000138	Dword	00014F7F	
Subsystem	0000013C	Word	0002	Windows GUI
DllCharacteristics	0000013E	Word	8000	Click here
SizeOfStackReserve	00000140	Dword	00040000	
SizeOfStackCommit	00000144	Dword	00011000	
SizeOfHeapReserve	00000148	Dword	00100000	
SizeOfHeapCommit	0000014C	Dword	00001000	
LoaderFlags	00000150	Dword	00000000	
NumberOfRvaAndSizes	00000154	Dword	00000010	

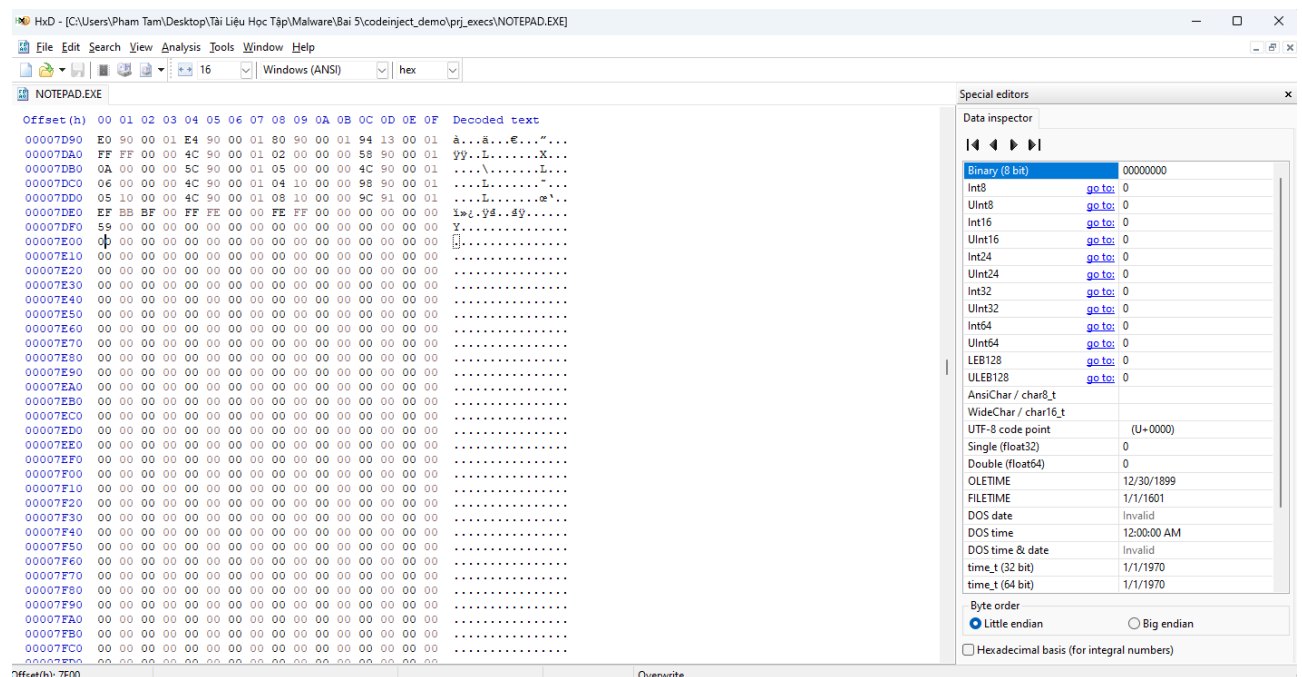
- Sau đó lưu lại, mở file NOTEPAD.EXE lên kiểm tra kết quả



Bài thực hành 5: Bằng cách không tạo thêm vùng nhớ mở rộng vào tập tin PE, tận dụng vùng nhớ trống để chèn chương trình cần chèn trên tập tin Notepad và calc.

NOTEPAD.EXE:

- Chèn vào file PE tại Offset 0x00007E00:



Ta có địa chỉ MessageBox là : Z= 68120001

- Công thức tính X là :

$$0x00007E00 - 0x00007C00 = X - 0x00009000$$

$$\text{Vậy } X = 0x01009240$$

$$\text{Vậy } Y = 0x01009290$$

- Ta có `old_entry_point = 0x0100739D`; `jmp_instruction_VA = 0x01009214`

$$\text{Vậy } \text{relative_VA} = 0xFFFFE184$$

- Mở CFF Explorer của NOTEPAD.EXE vào Optional Header chỉnh sửa giá trị của `AddressOfEntryPoint` thành `0x00009200`

CFF Explorer VIII - [NOTEPAD.EXE]

File Settings ?

NOTEPAD.EXE

File: NOTEPAD.EXE

- Dos Header
- Nt Headers
 - File Header
 - Optional Header
 - Data Directories [x]
- Section Headers [x]
- Import Directory
- Resource Directory
- Debug Directory
- Address Converter
- Dependency Walker
- Hex Editor
- Identifier
- Import Adder
- Quick Disassembler
- Rebuilder
- Resource Editor
- UPX Utility

Member	Offset	Size	Value	Meaning
Magic	000000F8	Word	010B	PE32
MajorLinkerVersion	000000FA	Byte	07	
MinorLinkerVersion	000000FB	Byte	0A	
SizeOfCode	000000FC	Dword	00007800	
SizeOfInitializedData	00000100	Dword	0000A600	
SizeOfUninitializedData	00000104	Dword	00000000	
AddressOfEntryPoint	00000108	Dword	00009200	.data
BaseOfCode	0000010C	Dword	00001000	
BaseOfData	00000110	Dword	00009000	
ImageBase	00000114	Dword	01000000	
SectionAlignment	00000118	Dword	00001000	
FileAlignment	0000011C	Dword	00000200	
MajorOperatingSystemVersion	00000120	Word	0005	
MinorOperatingSystemVersion	00000122	Word	0001	
MajorImageVersion	00000124	Word	0005	
MinorImageVersion	00000126	Word	0001	
MajorSubsystemVersion	00000128	Word	0004	
MinorSubsystemVersion	0000012A	Word	0000	
Win32VersionValue	0000012C	Dword	00000000	
SizeOfImage	00000130	Dword	00014000	
SizeOfHeaders	00000134	Dword	00000400	
Checksum	00000138	Dword	00014F7F	
Subsystem	0000013C	Word	0002	Windows GUI
DllCharacteristics	0000013E	Word	8000	Click here
SizeOfStackReserve	00000140	Dword	00040000	
SizeOfStackCommit	00000144	Dword	00011000	
SizeOfHeapReserve	00000148	Dword	00100000	
SizeOfHeapCommit	0000014C	Dword	00001000	
LoaderFlags	00000150	Dword	00000000	
NumberOfRvaAndSizes	00000154	Dword	00000010	

- Sử dụng HxD để chèn đoạn mã với giá trị Caption và Text

```
push 0 ; 6a 00
push Caption ; 68 40920001
push Text ; 68 90920001
push 0 ; 6a 00
call [MessageBoxW] ; ff15 68120001
jmp Original_Entry_point; e9 84E1FFFF
```

- Chèn giá trị Text là Nhóm 5 ; giá trị Caption là MSSV của thành viên nhóm

HxD - [C:\Users\Pham Tam\Desktop\Tài Liệu Học Tập\Malware\Bai 5\codeinject_demo\prj_execs\notepad.exe]

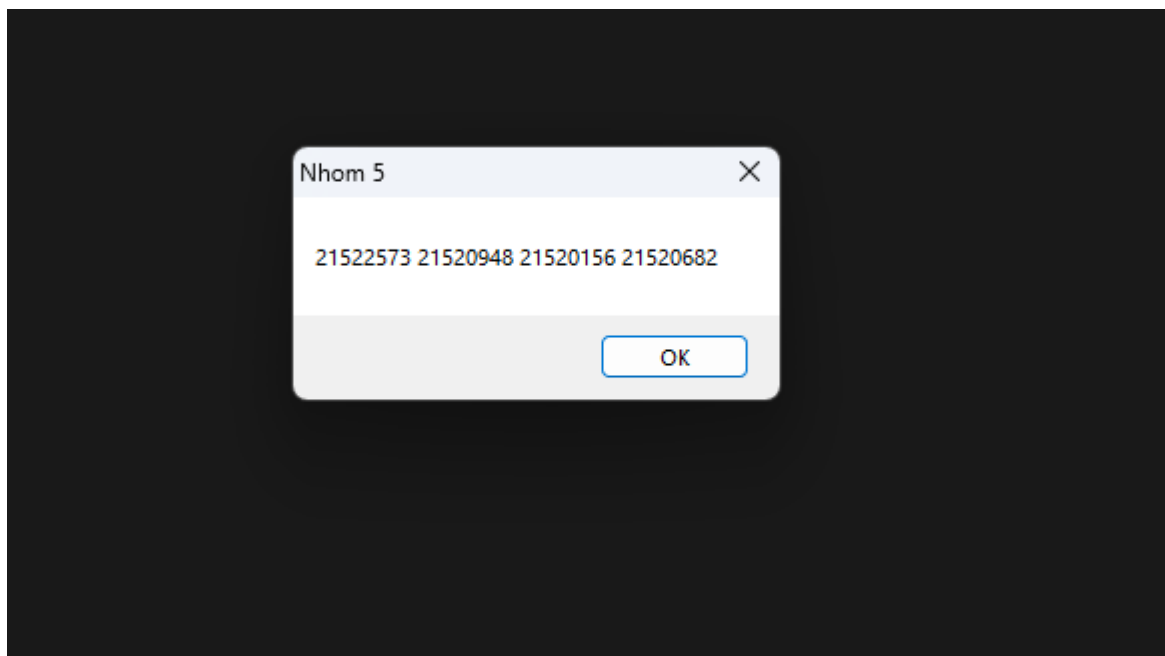
File Edit Search View Analysis Tools Window Help

16 Windows (ANSI) hex

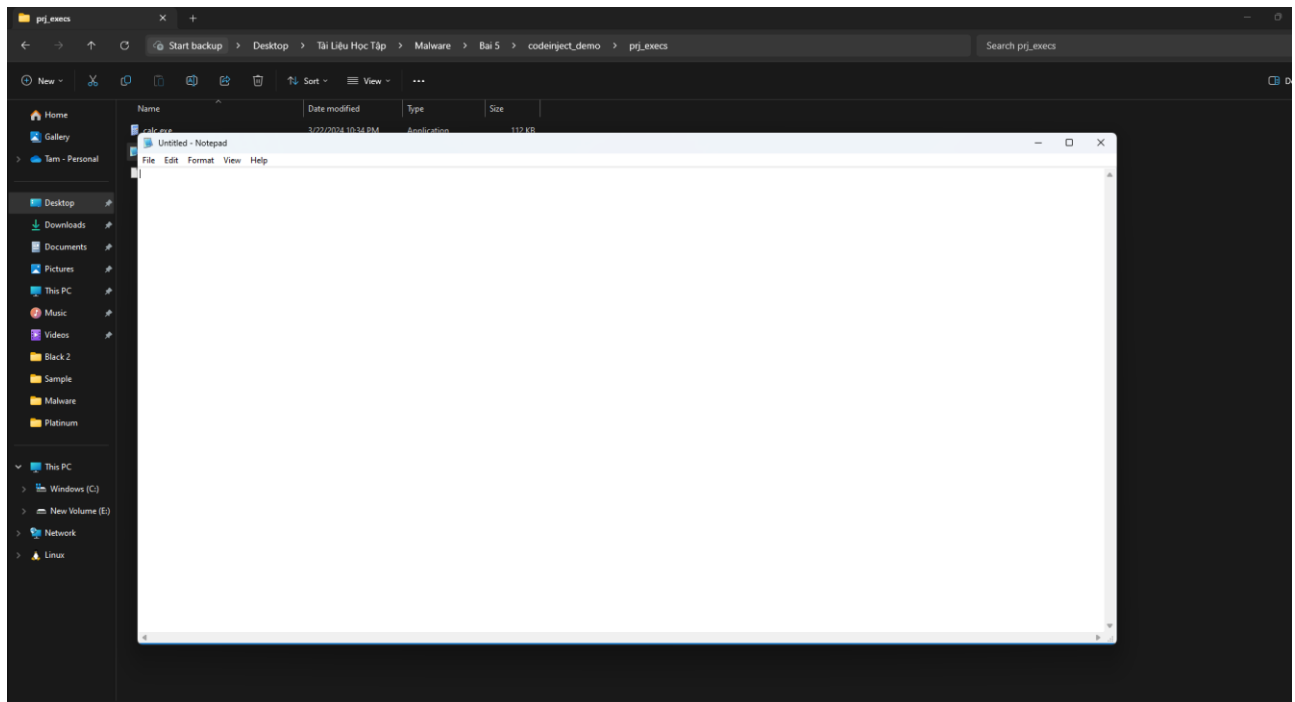
NOTEPAD.EXE

Offset(h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	Decoded text
00007D20	6C	90	00	01	70	90	00	01	74	90	00	01	84	90	00	01	l...p...t.....
00007D30	88	90	00	01	8C	90	00	01	90	90	00	01	94	90	00	01	...t.....
00007D40	98	90	00	01	9C	90	00	01	A0	90	00	01	A8	90	00	01	...t.....
00007D50	A4	90	00	01	AC	90	00	01	B0	90	00	01	B4	90	00	01	...t.....
00007D60	B8	90	00	01	BC	90	00	01	C0	90	00	01	C4	90	00	01	...t.....
00007D70	78	90	00	01	7C	90	00	01	C8	90	00	01	CC	90	00	01	...t.....
00007D80	D0	90	00	01	D4	90	00	01	D8	90	00	01	DC	90	00	01	...t.....
00007D90	E0	90	00	01	E4	90	00	01	80	90	00	01	94	13	00	01	...t.....
00007DA0	FF	FF	00	00	4C	90	00	01	02	00	00	00	58	90	00	01	...t.....
00007DB0	0A	00	00	00	5C	90	00	01	05	00	00	00	4C	90	00	01	...t.....
00007DC0	06	00	00	00	4C	90	00	01	04	10	00	00	98	90	00	01	...t.....
00007DD0	05	10	00	00	4C	90	00	01	08	10	00	00	9C	91	00	01	...t.....
00007DE0	EF	BB	BF	00	FF	FE	00	00	FE	FF	00	00	00	00	00	00	...t.....
00007DF0	59	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	...t.....
00007E00	6A	00	68	90	92	00	01	68	40	92	00	01	6A	00	FF	15	...t.....
00007E10	68	12	00	01	E9	84	E1	FF	FF	00	00	00	00	00	00	00	...t.....
00007E20	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	...t.....
00007E30	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	...t.....
00007E40	32	00	31	00	35	00	32	00	32	00	35	00	37	00	33	00	2.1.5.2.2.5.7.3.
00007E50	20	00	32	00	31	00	35	00	32	00	30	00	39	00	34	00	.2.1.5.2.0.9.4.
00007E60	38	00	20	00	32	00	31	00	35	00	32	00	30	00	31	00	8. .2.1.5.2.0.1.
00007E70	35	00	36	00	20	00	32	00	31	00	35	00	32	00	30	00	5.6. .2.1.5.2.0.
00007E80	36	00	38	00	32	00	00	00	00	00	00	00	00	00	00	00	6.8.2.....
00007E90	4E	00	68	00	6F	00	6D	00	20	00	35	00	00	00	00	00	N.h.o.m. .5....
00007EA0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00007EB0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00007EC0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00007ED0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00007EE0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00007EF0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00007F00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00

- Kết quả MessageBox:

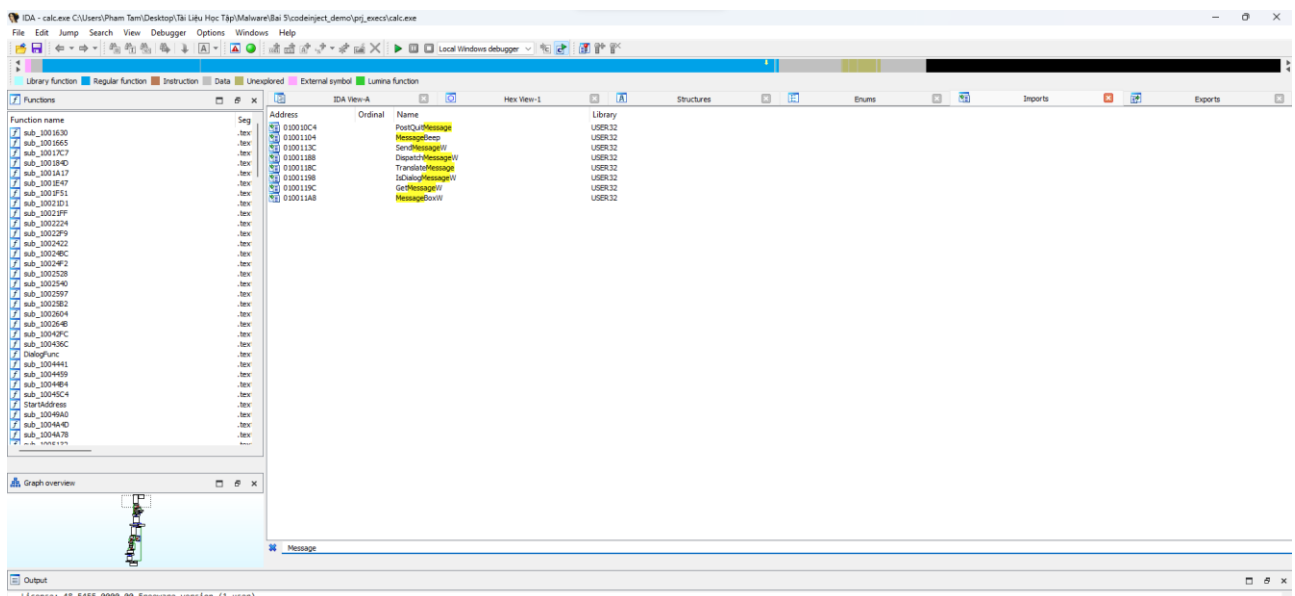


- Jump lại Notepad



CALC.EXE

- Mở IDA Pro và kiểm tra địa chỉ của MessageBox. Ta thấy MessageBoxW ở địa chỉ 0x010011A8



Vậy Z = 8A110001

- Mở CFF Explorer đọc nội dung của Calc.exe kiểm tra giá trị của AddressOfEntryPoint và Image Base trong Optional Header. Có giá trị lần lượt là 0x00012475 và 0x01000000

CFF Explorer VIII - [calc.exe]

Settings ?

calc.exe

Member	Offset	Size	Value	Meaning
Magic	00000108	Word	010B	PE32
MajorLinkerVersion	0000010A	Byte	07	
MinorLinkerVersion	0000010B	Byte	00	
SizeOfCode	0000010C	Dword	00012800	
SizeOfInitializedData	00000110	Dword	00009C00	
SizeOfUninitializedData	00000114	Dword	00000000	
AddressOfEntryPoint	00000118	Dword	00012475	.text
BaseOfCode	0000011C	Dword	00001000	
BaseOfData	00000120	Dword	00014000	
ImageBase	00000124	Dword	01000000	
SectionAlignment	00000128	Dword	00001000	
FileAlignment	0000012C	Dword	00000200	
MajorOperatingSystemVersion	00000130	Word	0005	
MinorOperatingSystemVersion	00000132	Word	0001	
MajorImageVersion	00000134	Word	0005	
MinorImageVersion	00000136	Word	0001	
MajorSubsystemVersion	00000138	Word	0004	
MinorSubsystemVersion	0000013A	Word	0000	
Win32VersionValue	0000013C	Dword	00000000	
SizeOfImage	00000140	Dword	0001F000	
SizeOfHeaders	00000144	Dword	00000400	
Checksum	00000148	Dword	0001D7FC	
Subsystem	0000014C	Word	0002	Windows GUI
DllCharacteristics	0000014E	Word	8000	Click here
SizeOfStackReserve	00000150	Dword	00040000	
SizeOfStackCommit	00000154	Dword	00001000	
SizeOfHeapReserve	00000158	Dword	00100000	
SizeOfHeapCommit	0000015C	Dword	00001000	
LoaderFlags	00000160	Dword	00000000	
NumberOfRvaAndSizes	00000164	Dword	00000010	

- Cộng 2 giá trị vừa tìm được ta được VA = 0x01012475

- Sử dụng HxD ta thấy cuối file calc.exe có khoảng trống. Chèn vào khoảng trống này, bắt đầu từ offset 0x0001BF50

Chèn giá trị Text ở offset 0x0001BF70

Chèn giá trị Caption ở offset 0x0001BFC0

```

Offset(h) 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F Decoded text
0001BD30 74 00 69 00 6E 00 75 00 65 00 2E 00 65 00 54 00 t.i.n.u.e.e.T.
0001BD40 68 00 65 00 20 00 72 00 65 00 71 00 75 00 65 00 h.e..r.e.q.u.e.
0001BD50 73 00 74 00 65 00 64 00 20 00 66 00 75 00 6E 00 e.t.e.d..f.u.n.
0001BD60 63 00 74 00 69 00 6F 00 6E 00 20 00 6D 00 61 00 c.t.i.o.n..m.a.
0001BD70 79 00 20 00 74 00 61 00 6B 00 65 00 20 00 61 00 y..t.a.k.e..a.
0001BD80 20 00 76 00 65 00 72 00 79 00 20 00 6C 00 6F 00 .v.e.r.y..l.o.
0001BD90 6E 00 67 00 20 00 74 00 69 00 6D 00 65 00 20 00 n.g..t.i.m.e..
0001BDA0 74 00 6F 00 20 00 63 00 6F 00 6D 00 70 00 6C 00 t.o..c.o.m.p.l.
0001BDB0 65 00 74 00 65 00 2E 00 00 00 44 00 6F 00 20 00 e.t.e....D.O..
0001BDC0 79 00 6F 00 75 00 20 00 77 00 61 00 6E 00 74 00 y.o.u..w.a.n.t..
0001BDD0 20 00 74 00 6F 00 20 00 61 00 62 00 6F 00 72 00 .t.o..a.b.o.r..
0001BDE0 74 00 20 00 74 00 68 00 65 00 20 00 6F 00 70 00 t..t.h.e..o.p.
0001BDF0 65 00 72 00 61 00 74 00 69 00 6F 00 6E 00 20 00 e.x.e.c.u.t.i.o.n..
0001BE00 6E 00 6F 00 77 00 3F 00 08 00 63 00 61 00 6C 00 n.o.w..t..c.a.l.
0001BE10 63 00 2E 00 68 00 6C 00 70 00 00 00 00 00 00 00 c..h.i.p.....
0001BE20 16 00 43 00 61 00 6E 00 6E 00 6F 00 74 00 20 00 .C.a.n.n.o.t..
0001BE30 6F 00 70 00 65 00 6E 00 20 00 43 00 6C 00 69 00 o.p.e.n..C.l.i.
0001BE40 70 00 62 00 6F 00 61 00 72 00 64 00 2E 00 54 00 p.b.o.a.r.d...T.
0001BE50 54 00 68 00 65 00 72 00 65 00 20 00 69 00 73 00 T.h.e.r.e..i.s.
0001BE60 20 00 6E 00 74 00 20 00 65 00 6E 00 6F 00 .n.o.t..e.n.o.
0001BE70 75 00 67 00 68 00 20 00 6D 00 65 00 6D 00 6F 00 u.g.h..m.e.m.o.
0001BE80 72 00 79 00 20 00 66 00 6F 00 72 00 20 00 64 00 r.y..f.o.r..d.
0001BE90 61 00 74 00 61 00 2E 00 0D 00 43 00 6C 00 6F 00 a.t.a....C.l.o.
0001BEA0 73 00 65 00 20 00 6F 00 6E 00 65 00 20 00 6F 00 s.e..o.n.e..o.
0001BEB0 72 00 20 00 6D 00 6F 00 72 00 65 00 20 00 70 00 r..m.o.r.e..p.
0001BEC0 72 00 6F 00 47 00 72 00 61 00 6D 00 73 00 2C 00 r.o.g.r.a.m.s..r.
0001BED0 20 00 61 00 6E 00 64 00 20 00 74 00 68 00 65 00 .a.n.d..t.h.e.
0001BEE0 6E 00 20 00 74 00 72 00 79 00 20 00 61 00 67 00 n..t.r.y..a.g.
0001BEF0 61 00 69 00 6E 00 2E 00 08 00 63 00 61 00 6C 00 a.i.n....C.a.l.
0001BF00 63 00 2E 00 63 00 68 00 6D 00 0A 00 43 00 61 00 C...c.h.m...C.a.
0001BF10 6C 00 63 00 75 00 6C 00 61 00 74 00 6F 00 72 00 l.c.u.l.a.t.o.r.
0001BF20 11 00 4E 00 6F 00 74 00 20 00 45 00 6E 00 6F 00 .N.o.t..E.n.o.
0001BF30 75 00 67 00 68 00 20 00 4D 00 65 00 6D 00 6F 00 u.g.h..M.e.m.o.
0001BF40 72 00 79 00 00 00 00 00 00 00 00 00 00 00 00 00 r.y.....
0001BF50 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0001BF60 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0001BF70 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0001BF80 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0001BF90 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0001BFA0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0001BFB0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0001BFC0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0001BFD0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0001BFE0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0001BFF0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
    
```

- Sử dụng CFF Explorer kiểm tra giá trị Raw Address và Virtual Address cần chèn trong phần Section Header lần lượt có giá trị là 0x00013600 và 0x00016000

Name	Virtual Size	Virtual Address	Raw Size	Raw Address	Reloc Address	Linenumbers	Relocations N...	Linenumbers ..	Characteristics
00000238	00000240	00000244	00000248	0000024C	00000250	00000254	00000258	0000025A	0000025C
Byte[8]	Dword	Dword	Dword	Dword	Dword	Word	Word	Dword	
.text	000126B0	00001000	00012800	00000400	00000000	00000000	0000	0000	60000020
.data	0000101C	00014000	00000A00	00012C00	00000000	00000000	0000	0000	C0000040
.rsrc	00000960	00016000	00008A00	00013600	00000000	00000000	0000	0000	40000040

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	Ascii
00008900	63	00	2E	00	63	00	68	00	6D	00	0A	00	43	00	61	00	c...c.h.m...C.a.
00008910	6C	00	63	00	75	00	6C	00	61	00	74	00	6F	00	72	00	l.c.u.l.a.t.o.r.
00008920	11	00	4E	00	6F	00	74	00	20	00	45	00	6E	00	6F	00	.N.o.t..E.n.o.
00008930	75	00	67	00	68	00	20	00	4D	00	65	00	6D	00	6F	00	u.g.h..M.e.m.o.
00008940	72	00	79	00	00	00	00	00	00	00	00	00	00	00	00	00	r.y.....
00008950	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00008960	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00008970	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00008980	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00008990	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000089A0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000089B0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000089C0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000089D0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000089E0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000089F0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00

- Theo công thức tính X:

$$0x0001BF70 - 0x00013600 = X - 0x00016000$$

Vậy $X = 0x101E970$ (Đã cộng ImageBase)

- Theo công thức tính $Y = 0x101E9C0$

$$\text{New_entry_point} = 0x0001BF50 - 0x00013600 + 0x00016000 = 0x0001E950$$

Cộng thêm ImageBase. Vậy $\text{Address_Entry_Point} = 0101E950$

$$\text{Jmp_instruction_VA} = 0x0101E950 + 0x14 = 0x0101E964$$

Như ta đã kiểm tra old_entry_point có giá trị là $0x01012475$

$$\text{Vậy relative_VA} = 0x01012475 - 5 - 0x0101E964 = 0xFFFF3B0C$$

- Code Assembly hoàn chỉnh :

```
push 0           ; 6a 00
push Caption     ; 68 C0E90101
push Text        ; 68 70E90101
push 0           ; 6a 00
call [MessageBoxW] ; ff15 A8110001
jmp Original_Entry_Point ; e9 0C3BFFFF
```

- Thay đổi giá trị $\text{AddressOfEntryPoint}$ trong CFF Explorer thành $0001E950$

CFF Explorer VIII - [calc.exe]

File Settings ?

calc.exe calc.exe

File: calc.exe

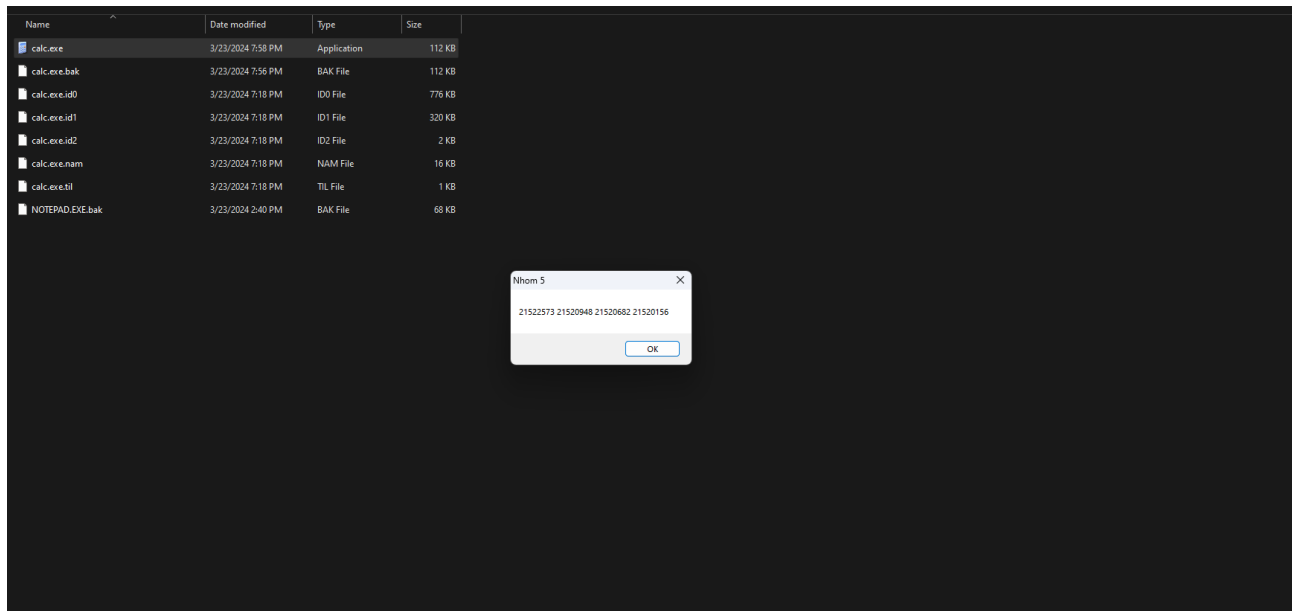
- Dos Header
- Nt Headers
- File Header
- Optional Header
 - Data Directories [x]
- Section Headers [x]
- Import Directory
- Resource Directory
- Debug Directory
- Address Converter
- Dependency Walker
- Hex Editor
- Identifier
- Import Addr
- Quick Disassembler
- Rebuilder
- Resource Editor
- UPX Utility

Member	Offset	Size	Value	Meaning
Magic	00000108	Word	010B	PE32
MajorLinkerVersion	0000010A	Byte	07	
MinorLinkerVersion	0000010B	Byte	00	
SizeOfCode	0000010C	Dword	00012800	
SizeOfInitializedData	00000110	Dword	00009C00	
SizeOfUninitializedData	00000114	Dword	00000000	
AddressOfEntryPoint	00000118	Dword	0001E950	.rsrc
BaseOfCode	0000011C	Dword	00001000	
BaseOfData	00000120	Dword	00014000	
ImageBase	00000124	Dword	01000000	
SectionAlignment	00000128	Dword	00001000	
FileAlignment	0000012C	Dword	00000200	
MajorOperatingSystemVersion	00000130	Word	0005	
MinorOperatingSystemVersion	00000132	Word	0001	
MajorImageVersion	00000134	Word	0005	
MinorImageVersion	00000136	Word	0001	
MajorSubsystemVersion	00000138	Word	0004	
MinorSubsystemVersion	0000013A	Word	0000	
Win32VersionValue	0000013C	Dword	00000000	
SizeOfImage	00000140	Dword	0001F000	
SizeOfHeaders	00000144	Dword	00000400	
Checksum	00000148	Dword	0001D7FC	
Subsystem	0000014C	Word	0002	Windows GUI
DllCharacteristics	0000014E	Word	8000	Click here
SizeOfStackReserve	00000150	Dword	00040000	
SizeOfStackCommit	00000154	Dword	00001000	
SizeOfHeapReserve	00000158	Dword	00100000	
SizeOfHeapCommit	0000015C	Dword	00001000	
LoaderFlags	00000160	Dword	00000000	
NumberOfRvaAndSizes	00000164	Dword	00000010	

- Nhập giá trị trong HxD

```
Offset(h) 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F Decoded text
0001BDB0 65 00 74 00 65 00 2E 00 0D 00 44 00 6F 00 20 00 e.e....D.o. .
0001BDC0 79 00 6F 00 75 00 20 00 77 00 61 00 6E 00 74 00 y.o.u..w.a.n.t.
0001BDD0 20 00 74 00 6F 00 20 00 61 00 62 00 6F 00 72 00 .t.o..a.b.o.r.
0001BDE0 74 00 20 00 74 00 68 00 65 00 20 00 6F 00 70 00 t..t.h.e..o.p.
0001BDF0 65 00 72 00 61 00 74 00 69 00 6F 00 6E 00 20 00 e.r.a.t.i.o.n. .
0001BE00 6E 00 6F 00 77 00 3F 00 08 00 63 00 61 00 6C 00 n.o.w?...C.a.l.
0001BE10 63 00 2E 00 68 00 6C 00 70 00 00 00 00 00 00 00 C..h.l.p.....
0001BE20 16 00 43 00 61 00 6E 00 6E 00 6F 00 74 00 20 00 ..C.a.n.n.o.t. .
0001BE30 6F 00 70 00 65 00 6E 00 20 00 43 00 6C 00 69 00 o.p.e.n..C.l.i.
0001BE40 70 00 62 00 6F 00 61 00 72 00 64 00 2E 00 54 00 p.b.o.a.r.d...T.
0001BE50 54 00 68 00 65 00 72 00 65 00 20 00 69 00 73 00 T.h.e.r.e..i.s.
0001BE60 20 00 6E 00 6F 00 74 00 20 00 65 00 6E 00 6F 00 .n.o.t..e.n.o.
0001BE70 75 00 67 00 68 00 20 00 6D 00 65 00 6D 00 6F 00 u.g.h..m.e.m.o.
0001BE80 72 00 79 00 20 00 66 00 6F 00 72 00 20 00 64 00 r.y..f.o.r..d.
0001BE90 61 00 74 00 61 00 2E 00 0D 00 43 00 6C 00 6F 00 a.t.a....C.l.o.
0001BEA0 73 00 65 00 20 00 6F 00 6E 00 65 00 20 00 6F 00 s.e..o.n.e..o.
0001BEB0 72 00 20 00 6D 00 6F 00 72 00 65 00 20 00 70 00 r..m.o.r.e..p.
0001BEC0 72 00 6F 00 67 00 72 00 61 00 6D 00 73 00 2C 00 r.o.g.r.a.m.s.,.
0001BED0 20 00 61 00 6E 00 64 00 20 00 74 00 68 00 65 00 .a.n.d..t.h.e.
0001BEE0 6E 00 20 00 74 00 72 00 79 00 20 00 61 00 67 00 n..t.r.y..a.g.
0001BEF0 61 00 69 00 6E 00 2E 00 08 00 63 00 61 00 6C 00 a.i.n....C.a.l.
0001BF00 63 00 2E 00 63 00 68 00 6D 00 0A 00 43 00 61 00 C..c.h.m...C.a.
0001BF10 6C 00 63 00 75 00 6C 00 61 00 74 00 6F 00 72 00 l.c.u.l.a.t.o.r.
0001BF20 11 00 4E 00 6F 00 74 00 20 00 45 00 6E 00 6F 00 .N.o.t..e.n.o.
0001BF30 75 00 67 00 68 00 20 00 4D 00 65 00 6D 00 6F 00 u.g.h..M.e.m.o.
0001BF40 72 00 79 00 00 00 00 00 00 00 00 00 00 00 00 00 r.y.....
0001BF50 6A 00 68 00 E9 01 01 68 70 E9 01 01 6A 00 FF 15 j.h.á..h.pé..j.ý.
0001BF60 A8 11 00 01 E9 0C 3B FF FF 00 00 00 00 00 00 00 00 ..é..ý.....
0001BF70 32 00 31 00 35 00 32 00 32 00 35 00 37 00 33 00 2.1.5.2.2.5.7.3.
0001BF80 20 00 32 00 31 00 35 00 32 00 30 00 39 00 34 00 .2.1.5.2.0.9.4.
0001BF90 38 00 20 00 32 00 31 00 35 00 32 00 30 00 36 00 8..2.1.5.2.0.6.
0001BFA0 38 00 32 00 20 00 32 00 31 00 35 00 32 00 30 00 8.2..2.1.5.2.0.
0001BFB0 41 00 35 00 36 00 00 00 00 00 00 00 00 00 00 00 1.5.6.....
0001BFC0 3E 00 68 00 6F 00 6D 00 20 00 35 00 00 00 00 00 N.h.o.m..s.....
0001BFD0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0001BFE0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0001BFF0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
```

- Lưu lại và kiểm tra file calc.exe



- Jump lại calc.exe

