

## Description

What does `asm1(0x2e0)` return? Submit the flag as a hexadecimal value (starting with '0x').

NOTE: Your submission for this question will NOT be in the normal flag format. [Source](#)

## Hints

- assembly [conditions](#)

## Solución

```
kali@kali: ~/shared/notas-seguridad-redes2024/picoCTF/categorias/reversing/parte_03/asm1
File Actions Edit View Help
GNU nano 8.1 test.S *
0000
ffffff
registers
asm1:
<+0>: push ebp
<+1>: mov ebp,esp
<+3>: cmp DWORD PTR [ebp+0x8],0x3fb
<+10>: jg 0x512 <asm1+37>
<+12>: cmp DWORD PTR [ebp+0x8],0x280
<+19>: jne 0x50a <asm1+29>
<+21>: mov eax,DWORD PTR [ebp+0x8]
^G Help ^O Write Out ^F Where Is ^K Cut ^T Execute ^C Location
^X Exit ^R Read File ^\ Replace ^U Paste ^J Justify ^_ Go To Line
(kali@kali)-[~/../categorias/reversing/parte_03/asm1]
$
```

```
GNU nano 8.1 test.S *
0000
[ebp]
[ret]
[0x2e0]
ffffff
registers
asm1:
..
<+0>: push ebp
<+1>: mov ebp,esp
<+3>: cmp DWORD PTR [ebp+0x8],0x3fb
<+10>: jg 0x512 <asm1+37>
```

```
(kali@kali)-[~/../categorias/reversing/parte_03/asm1]
$ cat test.S
STACK
```

0000

```
[
[ebp] <- esp <- ebp
[ret] <- ebp + 0x4
[0x2e0] <- ebp + 0x8
fffff
```

registers

[0x2d6] eax

asm1:

```
      <+0>:  push    ebp
      <+1>:  mov     ebp,esp

      <+3>:  cmp     DWORD PTR [ebp+0x8],0x3fb
..      <+10>:  jg      0x512 <asm1+37>
      <+12>:  cmp     DWORD PTR [ebp+0x8],0x280
      <+19>:  jne     0x50a <asm1+29>
      <+21>:  mov     eax,DWORD PTR [ebp+0x8]
      <+24>:  add     eax,0xa
      <+27>:  jmp     0x529 <asm1+60>
      <+29>:  mov     eax,DWORD PTR [ebp+0x8]
..      <+32>:  sub     eax,0xa
      <+35>:  jmp     0x529 <asm1+60>
      <+37>:  cmp     DWORD PTR [ebp+0x8],0x559
      <+44>:  jne     0x523 <asm1+54>
      <+46>:  mov     eax,DWORD PTR [ebp+0x8]
      <+49>:  sub     eax,0xa
      <+52>:  jmp     0x529 <asm1+60>
      <+54>:  mov     eax,DWORD PTR [ebp+0x8]
      <+57>:  add     eax,0xa

..      <+60>:  pop     ebp
      <+61>:  ret
```

└─(kali㉿kali)-[~/.../categorias/reversing/parte\_03/asm1]

└─\$ python3

Python 3.12.6 (main, Sep 7 2024, 14:20:15) [GCC 14.2.0] on linux

Type "help", "copyright", "credits" or "license" for more information.

```
>>> 0x2e0 > 0x3fb
```

False

```
>>> 0x2e0 > 0x280
```

True

```
>>> hex(0x2e0 - 0xa)
'0x2d6'
>>>
```

## Bandera

```
flag: 0x2d6
```

## Notas Adicionales

## Referencias

- [guide-to-assembly](#)