

Esercizio Epicode 29.11.2023

Identificare le istruzioni del seguente codice in Assembly x86

```
0x00001141 <+8>: mov EAX,0x20
0x00001148 <+15>: mov EDX,0x38
0x00001155 <+28>: add EAX, EDX
0x00001157 <+30>: mov EBP, EAX
0x0000115a <+33>: cmp EBP,0xa
0x0000115e <+37>: jge 0x1176 <main+61>
0x0000116a <+49>: mov eax,0x0
0x0000116f <+54>: call 0x1030 <printf@plt>
```

Tutti i valori dati prima dei comandi indicano gli indirizzi di memoria delle istruzioni nel programma

mov EAX, 0x20:

Carica il valore 32, in decimale, nel registro EAX.

mov EDX, 0x38:

Carica il valore 56, in decimale, nel registro EDX.

add EAX, EDX:

Somma il contenuto del registro EDX al registro EAX e memorizza il risultato in EAX. Quindi, EAX conterrà 88, in decimale.

mov EBP, EAX:

Carica il contenuto del registro EAX nel registro EBP.

cmp EBP, 0xa:

Confronta il contenuto del registro EBP con il valore decimale 10

jge 0x1176:

L'istruzione JGE "Jump if Greater Than or Equal" (Salta se Maggiore o Uguale), salta all'indirizzo 0x1176 se il valore in EBP è maggiore o uguale a 10 decimale.

Se la comparazione è falsa (il valore in EBP è inferiore a 10), esegue le seguenti istruzioni:

mov eax, 0x0:

Carica il valore 0 decimale nel registro EAX.

call 0x1030 <printf@plt>:

Chiama la funzione situata all'indirizzo 0x1030, che sembra essere la funzione **printf**.

Si tratta di una sequenza di istruzioni confronta un numero se maggiore o uguale a 10 e stampa qualcosa a schermo in base al risultato ottenuto.