

ESERCIZIO 3 SETTIMANA 10

mov EAX, 0x20: Questa istruzione sposta il valore esadecimale 0x20 (32 in decimale) nel registro EAX.

mov EDX, 0x38: Qui, il valore esadecimale 0x38 (56 in decimale) viene spostato nel registro EDX.

add EAX, EDX: Aggiunge il contenuto del registro EDX (56) al registro EAX (32) e memorizza il risultato in EAX.

mov EBP, EAX: Il contenuto di EAX, che ora è la somma di 32 e 56 (cioè 88 in decimale), viene copiato nel registro EBP.

cmp EBP, 0xa: Compara il contenuto di EBP con il valore esadecimale 0xA (10 in decimale).

jge 0x1176 <main+61>: Salta all'etichetta 0x1176 (<main+61>) se il risultato della comparazione precedente è maggiore o uguale (jge significa "jump if greater than or equal").

mov EAX, 0x0: Se la comparazione precedente non è soddisfatta (cioè se il valore in EBP è inferiore a 10), viene caricato il valore 0 nel registro EAX.

call 0x1030 <printf@plt>: Chiama la funzione di libreria printf, presumibilmente per stampare il contenuto di EAX, che ora è 0.

In breve, questo codice Assembly somma due numeri (32 e 56), li confronta con 10 e stampa 0 se la somma è inferiore a 10.