

Assembly x86

Obiettivo: Descrivere lo scopo di istruzioni di un codice date dall'esercizio.

Mov EAX, 0 x 20

Questa istruzione *Assegna* il valore **32** (0x20 in esadecimale) al **Registro EAX**.

Mov EDX, 0 x 38

Questa istruzione *Assegna* il valore **56** (0x38 in esadecimale) al **Registro EDX**.

Add EAX, EDX

L'istruzione fa la *Somma* dei valori di **EAX** e **EDX**, salvando il risultato in **EAX**.

Mov EBP, EAX

L'istruzione *Copia* il contenuto di **EAX** nel registro **EBP**.

Cmp EBP, 0 x a

In questo caso fa una *Comparazione* del valore di **EBP** con **10** (0xa in esadecimale).

Jge 0 x 1176 <main+61>

Con questa istruzione si indica che se **Cmp** è maggiore o uguale a **10** (Jump if Greater or Equal) *Salta* all'indirizzo 0x1176 altrimenti prosegue con la prossima istruzione. Il *Salto* avviene a un'istruzione che si trova a 61 byte di distanza dall'inizio della funzione "*main*".

Mov EAX, 0 x 0

L'istruzione *Copia* il valore **0** all'interno del **Registro EAX**.

Call 0 x 1030 <printf@plt>

Quest'ultima istruzione tenta una *Chiamata* ad una subroutine o a una funzione all'indirizzo 0x1030. Come si vede dal commento accanto probabilmente porta alla funzione *printf*.