

Interpretazione Assembly

♦ **0x00001141 <+8>: mov EAX,0x20**

Convertendo il valore 0x20 in decimale, si ottiene il valore 32. La stringa quindi vuole copiare il valore 32 nel registro EAX.

EAX = 32

♦ **0x00001148 <+15>: mov EDX,0x38**

Il valore esadecimale 0x38 corrisponde al numero 56. Deduciamo che vogliamo assegnare il valore 56 al registro EDX.

EDX = 56

♦ **0x00001155 <+28>: add EAX,EDX**

Si effettua una somma fra il valore assegnato al registro EAX al registro EDX, assegnandolo al registro EAX.

32 + 56 = 88

♦ **0x00001157 <+30>: mov EBP, EAX**

Copiamo il valore del registro EAX al registro EBP.

EBP = 88

♦ **0x0000115a <+33>: cmp EBP,0xa**

Compariamo il valore 0XA, ovvero 10, al valore del registro EBP

EBP = 88 diverso da 10

♦ **0x0000115e <+37>: jge 0x1176 <main+61>**

Saltiamo alla locazione 0x1176 <main+61> visto che nel registro EBP, ovvero la destinazione del comando cmp precedente, il valore assegnato (88) è maggiore di quello della sorgente (10)

♦ **0x0000116a <+49>: mov eax,0x0**

Assegniamo al registro EAX il valore 0 (che è 0x0 convertito in decimale)

EAX = 0

♦ **0x0000116f <+54>: call 0x1030 <printf@plt>**

Chiamiamo la classica funzione "Print"