**Assembly x86**

Traccia:

Nella lezione teorica del mattino, abbiamo visto i fondamenti del linguaggio Assembly. Dato il codice in Assembly per la CPU x86 allegato qui di seguito, identificare lo scopo di ogni istruzione, inserendo una descrizione per ogni riga di codice. Ricordate che i numeri nel formato 0xYY sono numeri esadecimali. Per convertirli in numeri decimali utilizzate pure un convertitore online, oppure la calcolatrice del vostro computer (per programmatori).

1. 0x00001141 <+8>: mov EAX,0x20
2. 0x00001148 <+15>: mov EDX,0x38
3. 0x00001155 <+28>: add EAX,EDX
4. 0x00001157 <+30>: mov EBP, EAX
5. 0x0000115a <+33>: cmp EBP,0xa
6. 0x0000115e <+37>: jge 0x1176 <main+61>
7. 0x0000116a <+49>: mov eax,0x0
8. 0x0000116f <+54>: call 0x1030 <printf@plt>

Istruzione: mov EAX,0x20

Descrizione: Carica il valore esadecimale 0x20 (32 in decimale) nel registro EAX.

Istruzione: mov EDX,0x38

Descrizione: Carica il valore esadecimale 0x38 (56 in decimale) nel registro EDX.

Istruzione: add EAX,EDX

Descrizione: Somma il valore contenuto nel registro EDX a quello nel registro EAX e salva il risultato in EAX.

Istruzione: mov EBP,EAX

Descrizione: Copia il contenuto del registro EAX nel registro EBP.

Istruzione: cmp EBP,0xa

Descrizione: Confronta il valore nel registro EBP con 0xa (10 in decimale) e imposta i flag di stato CPU basati sul risultato.

Istruzione: jge 0x1176 <main+61>

Descrizione: Effettua un salto condizionale all'indirizzo 0x1176 (4470 in decimale) se il valore in EBP è maggiore o uguale a 10.

Istruzione: mov eax,0x0

Descrizione: Imposta il registro eax a 0x0 (0 in decimale).

Istruzione: call 0x1030 <printf@plt>

Descrizione: Chiama la funzione printf presente all'indirizzo 0x1030.