

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).

0x00001141 <+8>: mov EAX,0x20 \*: spostare il valore 32 nel registro EAX

0x00001148 <+15>: mov EDX,0x38: spostare il valore 56 nel registro EDX

0x00001155 <+28>: add EAX,EDX: somma valore registro EDX col registro EAX

0x00001157 <+30>: mov EBP, EAX: muovi registro EAX nel registro EBP

0x0000115a <+33>: cmp EBP,0xa: compara 10 a EBP che non e uguale a EBP

0x0000115e <+37>: jge 0x1176 <main+61>: significa che salta alla locazione specificata 0x1176 <main+61> se la destinazione è maggiore o uguale della sorgente nell'istruzione.

0x0000116a <+49>: mov eax,0x0: muovi 0 nel registro eax

0x0000116f <+54>: call 0x1030 <printf@plt>: chiama una funziona all'indirizzo di memoria 0x1030 specificatamente la funziona printf