

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.

- 0x00001141 <+8>: mov EAX,0x20
Carica il valore esadecimale 0x20 nel registro EAX.
- 0x00001148 <+15>: mov EDX,0x38
Carica il valore esadecimale 0x38 nel registro EDX.
- 0x00001155 <+28>: add EAX,EDX
Aggiunge il contenuto di EDX al contenuto di EAX e salva il risultato in EAX.
- 0x00001157 <+30>: mov EBP, EAX
Copia il valore di EAX nel registro EBP.
- 0x0000115a <+33>: cmp EBP,0xa
Compara il valore in EBP con il numero 10 (0xa in esadecimale).
- 0x0000115e <+37>: jge 0x1176 <main+61>
Salta all'indirizzo 0x1176 se il confronto precedente ha dato esito maggiore o uguale, altrimenti continua con l'istruzione successiva.
- 0x0000116a <+49>: mov eax,0x0
Carica il valore esadecimale 0 nel registro EAX.
- 0x0000116f <+54>: call 0x1030 <printf@plt>
Chiama la funzione printf situata all'indirizzo 0x1030. Questa istruzione è parte di un programma più grande che utilizza printf per stampare qualcosa a schermo.