

Assembly x86

L'esercizio chiede, dato il codice in Assembly x86 di seguito, di identificare lo scopo di ogni istruzione, inserendo una descrizione per ogni riga di codice.

0x00001141 <+8>: mov EAX,0x20 ---> sposta il valore 32 (0x20) nel registro EAX

0x00001148 <+15>: mov EDX,0x38 ---> sposta il valore 56 (0x38) nel registro EDX

0x00001155 <+28>: add EAX,EDX ---> somma i registri EDX e EAX

0x00001157 <+30>: mov EBP, EAX ---> sposta il registro EAX nel registro EBP

0x0000115a <+33>: cmp EBP,0xa ---> compara il valore 10 (0xa) con il valore del registro EBP

0x0000115e <+37>: jge 0x1176 <main+61> ---> salta alla memoria 0x1176 <main+61>

0x0000116a <+49>: mov eax,0x0 ---> sposta il valore 0 (0x0) nel registro EAX di fatto facendolo diventare 0

0x0000116f <+54>: call 0x1030 <printf@plt> ---> chiama la funzione <printf@plt>