



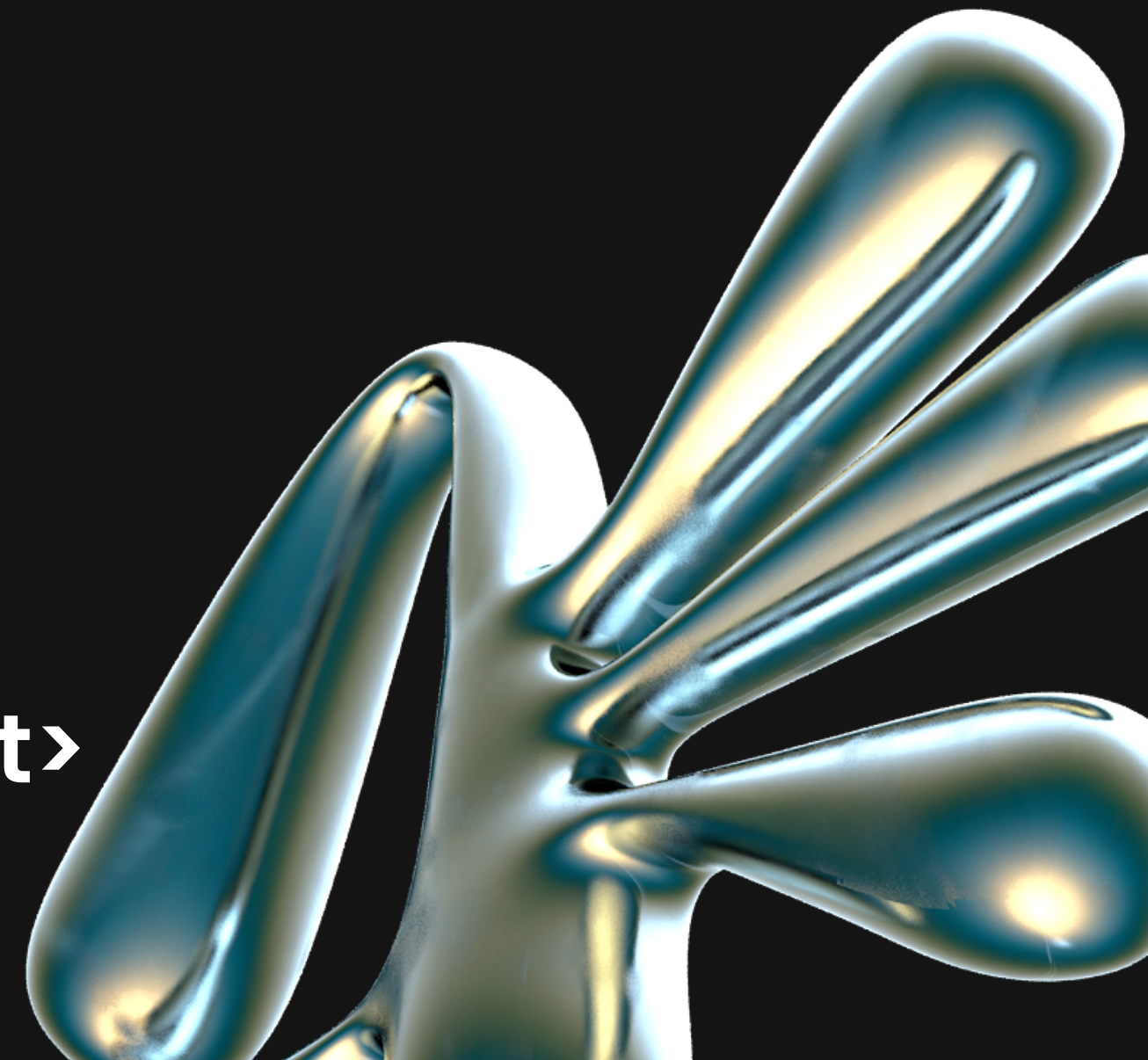
Assembly x86

Ernesto Robles

Assembly x86

Dato il seguente codice in Assembly per la CPU x86, verrà identificato lo scopo di ogni istruzione.

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



Assembly x86

mov EAX, 0x20: Carica il valore esadecimale 0x20 (32 decimale) nel registro EAX.

mov EDX, 0x38: Carica il valore esadecimale 0x38 (56 decimale) nel registro EDX.

add EAX, EDX: Aggiunge il contenuto di EDX a EAX e memorizza il risultato in EAX.

mov EBP, EAX: Muove il valore presente in EAX nel registro di base EBP.

cmp EBP, 0xa: Compara il valore presente in EBP con il numero 10 (0xa esadecimale).

jge 0x1176 <main+61>: Salta a 0x1176 (etichetta <main+61>) se la condizione ">=" è vera, ovvero se EBP è maggiore o uguale a 10.

mov eax, 0x0: Muove il valore 0 nel registro EAX.

call 0x1030 <printf@plt>: Chiama la funzione printf situata all'indirizzo 0x1030
“Procedure Linkage Table (PLT)” viene spesso utilizzata per stampare a schermo.

Assembly x86

Il codice Assembly prima analizzato è una parte di una funzione main di un programma scritto in linguaggio per l'architettura x86 che ha come scopo l'esecuzione di alcune operazioni aritmetiche e la stampa di un messaggio usando la funzione printf.