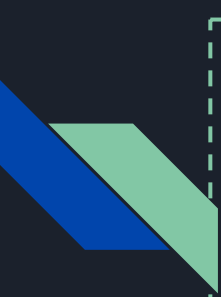


- 
1. **`mov EAX,0x20`**: Questa istruzione sposta il valore 0x20 (32 in decimale) nel registro EAX. Il registro EAX è comunemente utilizzato come registro generale per le operazioni aritmetiche e per memorizzare valori temporanei.
 2. **`mov EDX,0x38`**: Qui, il valore 0x38 (56 in decimale) viene spostato nel registro EDX. Come EAX, il registro EDX è utilizzato per memorizzare valori temporanei.
 3. **`add EAX,EDX`**: Aggiunge i contenuti dei registri EAX e EDX e memorizza il risultato in EAX. Quindi, ora il registro EAX contiene la somma di 32 e 56, che è 88 in decimale (o 0x58 in esadecimale).
 4. **`mov EBP, EAX`**: Copia il valore contenuto nel registro EAX nel registro EBP. Il registro EBP è spesso utilizzato come registro di base per l'accesso alle variabili locali e ai parametri delle funzioni.
 5. **`cmp EBP, 0xa`**: Compara il valore nel registro EBP con il numero 10 (0xa in esadecimale). La successiva istruzione condizionale **`jge`** (jump if greater than or equal) si baserà su questo confronto.
 6. **`jge 0x1176 <main+61>`**: Se il valore nel registro EBP è maggiore o uguale a 10, il programma salta all'indirizzo 0x1176 nel codice, che probabilmente corrisponde a un'altra parte della logica del programma.
 7. **`mov eax, 0x0`**: Se il valore nel registro EBP non è maggiore o uguale a 10, questa istruzione imposta il registro EAX a 0 (zero).
 8. **`call 0x1030 <printf@plt>`**: Chiama la funzione printf, che di solito viene utilizzata per stampare output formattato su console. La stringa di formato e gli argomenti appropriati saranno presenti nello stack o nei registri prima della chiamata a printf.