

## Report Assembly x86

Task: Identificare lo scopo di ogni istruzione, inserendo una descrizione per ogni riga di codice:

```
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>
```

## Istruzione Descrizione

<b>Mov EAX,0x20</b>  <b>Conversione numero esadecimale in decimale = 32</b>	<b>Crea una copia del valore 32 nel registro EAX</b>
<b>Mov EDX,0x38</b>  <b>Conversione numero esadecimale in decimale = 56</b>	<b>Crea una copia del valore 56 nel registro EDX</b>
<b>Add EAX,EDX</b>	<b>Addiziona il valore contenuto in EDX e EAX e salva il contenuto in registro EAX , il valore è diventa (32+56 = 88)</b>
<b>Mov EBP,EAX</b>	<b>Copia il contenuto del registro EAX nel registro EBP</b>
<b>Cmp EBP,0xa</b>  <b>Conversione numero esadecimale in decimale = 10</b>	<b>Compara il valore del registro EBP (88) con valore 10. Nel qual caso il destinatario sia maggiore rispetto alla sorgente nell'istruzione cmp, sia Zero Flag che Carry Flag hanno valore 0</b>
<b>Jge 0x1176 &lt;main+61&gt;</b>  <b>Conversione numero esadecimale in decimale = 4470</b>	<b>Se il valore in EBP è maggiore o uguale a 10 Salta a locazione specificata . In questo caso specifico essendo la destinazione maggiore della sorgente, salterà alla locazione 4470</b>

<b>Mov EAX,0x0</b>  <b>Conversione numero esadecimale in decimale = 0</b>	<b>Crea una copia del valore 0 nel registro EAX</b>
<b>Call 0x1030 &lt;printf@plt&gt;</b>  <b>Conversione numero esadecimale in decimale = 4144</b>	<b>Chiamata d'esecuzione del programma alla funzione "printf"</b>