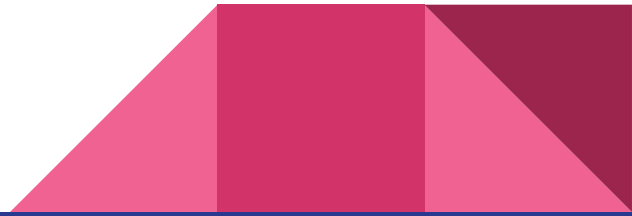
ASSEMBLY x86

S10L3

INDICE



INTRODUZIONE

Il linguaggio assembly x86 è un linguaggio di programmazione a basso livello impiegato principalmente per la programmazione di sistemi e l'embedded development su dispositivi che adottano architetture x86, come i computer basati su processori Intel o AMD. Le istruzioni assembly x86 sono strettamente legate all'architettura del processore e rappresentano direttamente le istruzioni macchina eseguite dal processore stesso.
La comprensione del linguaggio Assembly sarà utile per interpretare le istruzioni eseguite dalla CPU in una forma comprensibile per gli esseri umani.

TRACCIA

Nella lezione teorica del mattino, abbiamo visto i fondamenti del linguaggio Assembly. 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. Ricordate che i numeri nel formato 0xYY sono numeri esadecimali. Per convertirli in numeri decimali utilizzate pure un convertitore online, oppure la calcolatrice del vostro computer (per programmatori).

```
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>
```

SVOLGIMENTO

0x00001141 <+8>: mov EAX,0x20

converto 20 in decimale: 32
Poi “muovo” 32 in “EAX” ossia faccio un copia e incolla del 32 nel registro EAX.

0x00001148 <+15>: mov EDX,0x38

converto 38 in decimale: 56
Poi “muovo” 56 in “EDX” ossia faccio un copia e incolla del 56 nel registro EDX.

0x00001155 <+28>: add EAX,EDX

Sommo quello che è nel registro EDX con quello che è nel registro EAX.
Ossia 56 + 32 = 88

0x00001157 <+30>: mov EBP, EAX

Copio quello che è nel registro EAX nel registro EBP

SVOLGIMENTO

0x0000115a <+33>: cmp EBP,0xa

Trasformo in decimale la “a” e diventa 10. Poi “compare” (=confronto) il numero 10 con quello che è contenuto nel registro EBP.

0x0000115e <+37>: jge 0x1176 <main+61>

“Jump-greater-equal” ossia salta nell’istruzione “0x1176 <main+61>” se la condizione è maggiore uguale. Ossia se 88 è maggiore o uguale a 10. Effettivamente è così e quindi effettua il salto.

0x0000116a <+49>: mov eax,0x0

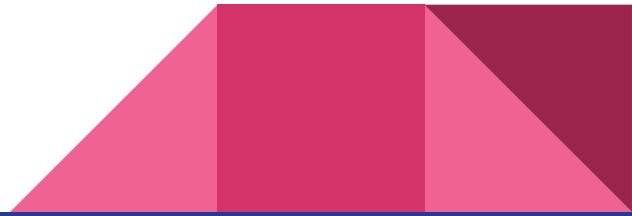
“Muove” il valore zero in eax rendendolo nullo.

0x0000116f <+54>: call 0x1030 <printf@plt>

“Chiama” la funzione “printf”

CONCLUSIONI

Possiamo affermare che il linguaggio assembly x86 è molto simile al linguaggio in “C”. Ovviamente molte cose ancora non le abbiamo studiate e quindi ancora non possiamo spiegare la parte di sinistra della traccia.



GRAZIE