



# ASM FOR ALL CHALLENGE

## Retro-Computer-Entertainment #06 - ASM for all Challenge

Inizio 15.03.2020

Terminé 14.04.2020

---

## L'Assembly per tutti

Da questo mese vorrei iniziare a lanciare una categoria diversa di sfide, infatti saranno dedicate al linguaggio macchina, ma in maniera tale che la sfida possa essere utilizzata come pungolo per impararlo da chi non lo conosce, mentre chi già mastica bene il linguaggio macchina potrà cimentarsi per misurarsi in una (o più) delle categoria experts in cui non si chiederà qualcosa in più rispetto al mero sviluppo di un programma che in una qualche maniera funzioni.

## Gestione della sfida

Verrà richiesto lo sviluppo di un programma in assembler (qualsiasi assembler supportato da computer 8/16 bit escluse le architetture 8086/8088). Verrà fornito il programma nel formato di poche righe in basic e/o un video che mostra il comportamento che dovrà avere il programma da fare.

I programmi inizialmente saranno molto semplici ma si complicheranno e/o andranno a coprire nuovi aspetti della programmazione assembly con l'andar delle sfide.

Coloro che raccoglieranno la sfida potranno coprire una o tutte le seguenti categorie:

- Sviluppo del programma con le stesse funzionalità;
- Sviluppo del programma con le stesse funzionalità e minor numero di istruzioni a design-time possibili;
- Sviluppo del programma con le stesse funzionalità e maggior velocità di esecuzione.

Naturalmente ognuna delle categorie va considerata all'interno della propria architettura.

## Regolamento

Nello sviluppare il codice, si chiede di evitare di utilizzare eventuali routine presenti in ROM su macchine specifiche (per esempio su C64 le routine del Kernal).

Si chiede di commentare bene il codice, un buon commento fa risparmiare il tempo che si perderebbe a capire cosa faccia un programma fatto da qualcun altro o peggio da se stessi.

Durante il periodo della sfida si incoraggia lo scambio di opinioni e pezzi di codice, anche in prospettiva del fatto che alcune persone si stanno approcciando a questo tipo di programmazione. Si chiede comunque di non pubblicare sin da subito il programma completo, sia per dar modo di imparare a chi non conosce perfettamente l'argomento sia per dare un piccolo tocco competitivo alla sfida.



I programmi Verranno raccolti e catalogati alla fine della sfida.

La durata della sfida è di 1 Mese, quindi se la data del documento di lancio della sfida riporta il giorno 15 (come in questo caso), la sfida terminerà il 14 del mese successivo (compreso).

I programmi potranno essere inviati a [Emanuele.Bonin71@Gmail.Com](mailto:Emanuele.Bonin71@Gmail.Com) accompagnati anche da due righe di spiegazione, sulla metodologia utilizzata o i trucchi utilizzati.

## Riempire & animare lo schermo con i colori

In questa prima sfida si chiede di riscrivere in asm il comportamento del programma in basic V2 per c64 riportato più sotto. Non è difficile capirne il funzionamento per la riproduzione su altri dialetti basic. In ogni caso vi è anche un video che ne mostra il comportamento.

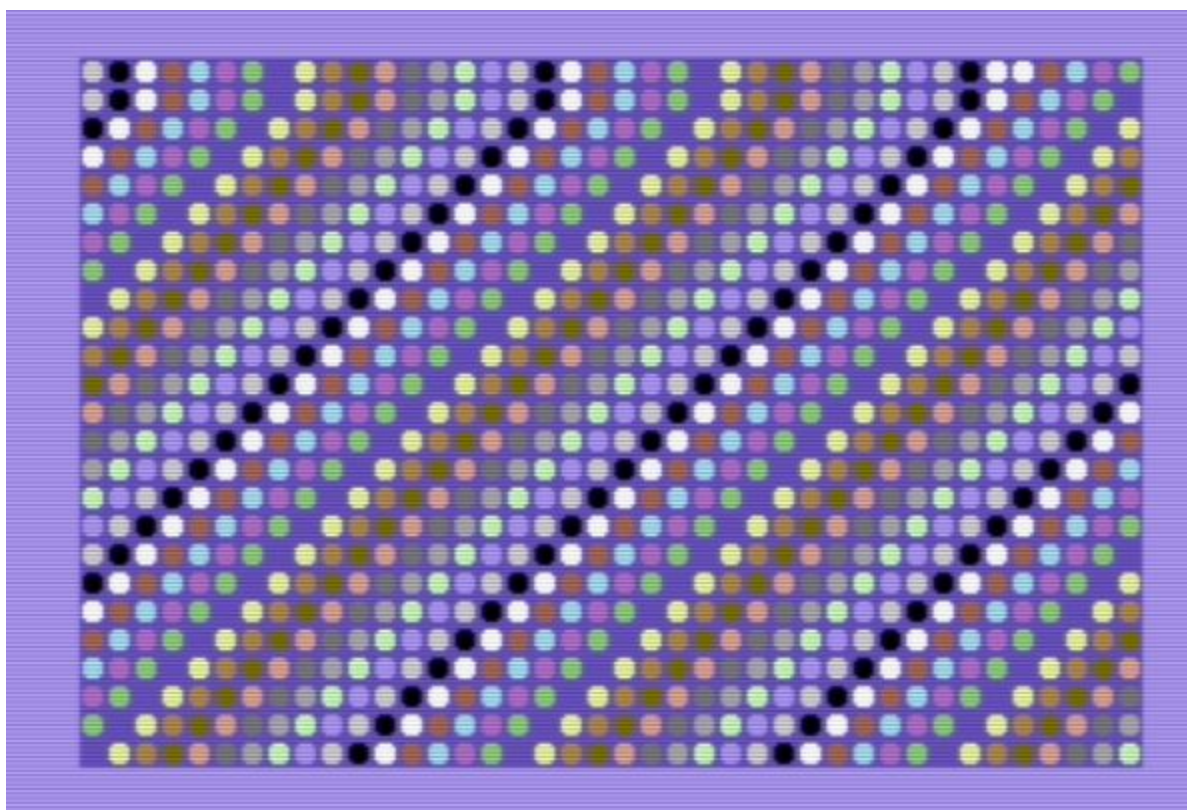
Non è necessario seguire pedissequamente lo sviluppo del programma in basic, l'importante è che il comportamento finale sia quello nel video allegato a questa challenge.

```

01 rem ASM FOR ALL Challenge #01
02 rem -----
05 rem 1) Riempire tutto lo schermo di caratteri
06 rem     nel c64 memoria video di norma risiede tra 1024 e 2032
10 for i = 1024 to 2032: poke i,81:next
20 rem 2) Calcolare un colore in base alla riga/colonna e assegnarlo
25 rem     al carattere di quella riga/colonna
28 rem     nel c64 si assegna il colore tramite la scrittura nella memoria
29 rem     colore che nel c64 risiede tra 55296 e 56295
30 for i = 55296 to 56295
35 rem     i colori nel c64 sono al massimo 16 (0-15) questa e' la ragione
38 rem     dell'and nel valore della poke, sebbene il c64 consideri comunque
39 rem     solo i 4 bit meno significativi del valore nella cella colore
40 poke i, ((r+c) and 15):c=c+1:if c=40 then c=0:r=r+1
50 next
55 rem 3) ciclo sulla mappa del colore incrementando il colore
58 rem     di ogni ogni cella video
60 for i = 55296 to 56295:c=peek(i):c=(c+1) and 15: if c=256 then c= 0
70 poke i,c:next
75 rem 4) ripetizione all'infinito del punto 3)
80 goto 60

```

Per coloro che non sapessero da dove iniziare vi consiglio, per coloro che si vogliono cimentare con l'assembly 6502, l'unità nel gruppo facebook RetroProgramming Italia "Bed Time Coding", in particolar modo la numero 7, dove si affronta il riempimento dello schermo. Altra unit molto utile quella sulla "Ludoprogrammazione" sempre nello stesso gruppo facebook.



Un momento dell'esecuzione del programma.