|  |  |  |
| --- | --- | --- |
| Altomare Andrea | Lab. Sistemi esperienza n. 10 | Classe 3^A 4/03/2015 |

**Assembly 8086: vettori**

**PROBLEMA**

Dati due vettori contenenti numeri diversi, copiare in un terzo vettore i numeri che si trovano in tutti e due quelli precedenti.

Vettori:

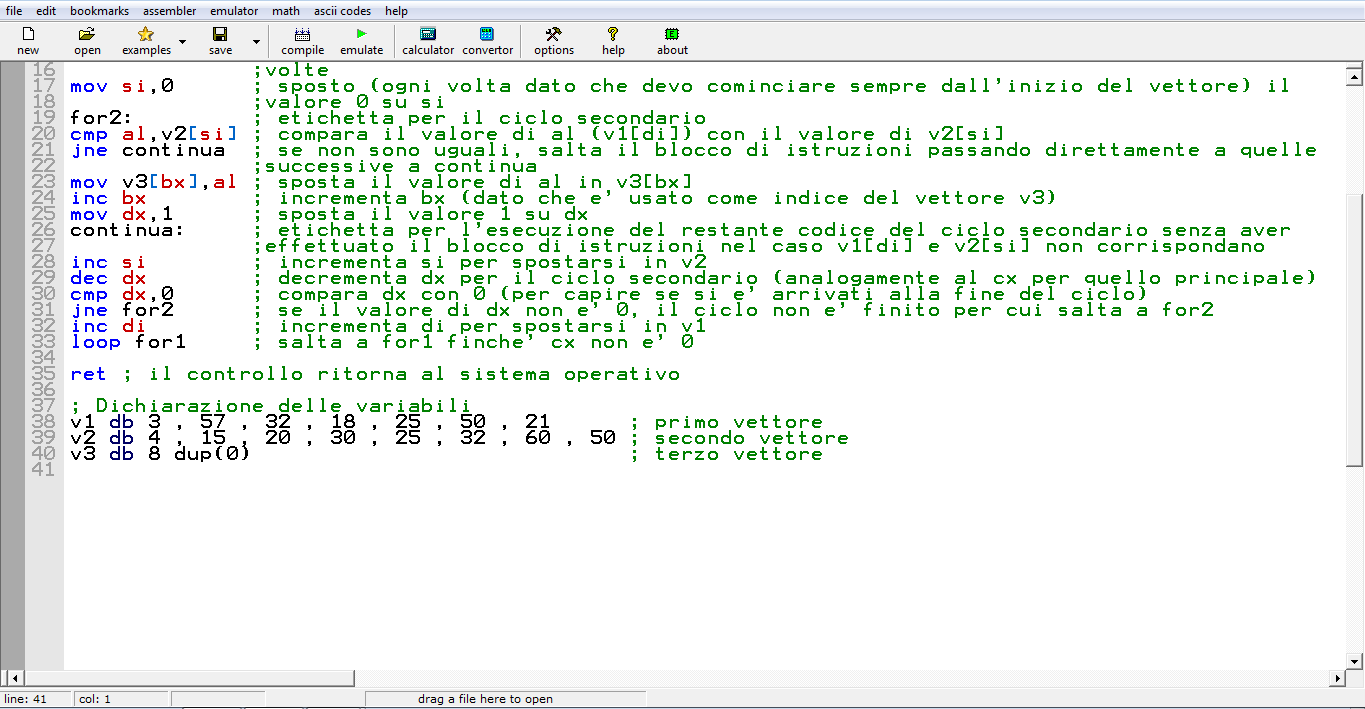
* *v1 db 3 , 57 , 32 , 18 , 25 , 50 , 21*
* *v2 db 4 , 15 , 20 , 30 , 25 , 32 , 60 , 50*
* *v3 db 8 dup(0)* (vettore contenente otto 0)

**REALIZZAZIONE DEL PROGRAMMA**

Per confrontare ciclicamente un valore del primo vettore (*v1*) con un valore del secondo (*v2*) ho bisogno di due cicli: il principale (quello più esterno) avrà il classico funzionamento con l’uso di cx come contatore e l’istruzione **loop** per decrementarlo e tornare (finché non è 0) all’etichetta di inizio ciclo, mentre per il ciclo secondario (quello più interno) ho dovuto “ricreare” manualmente (tramite l’utilizzo dei comandi adeguati) il loop (non usando, ovviamente, cx come contatore).

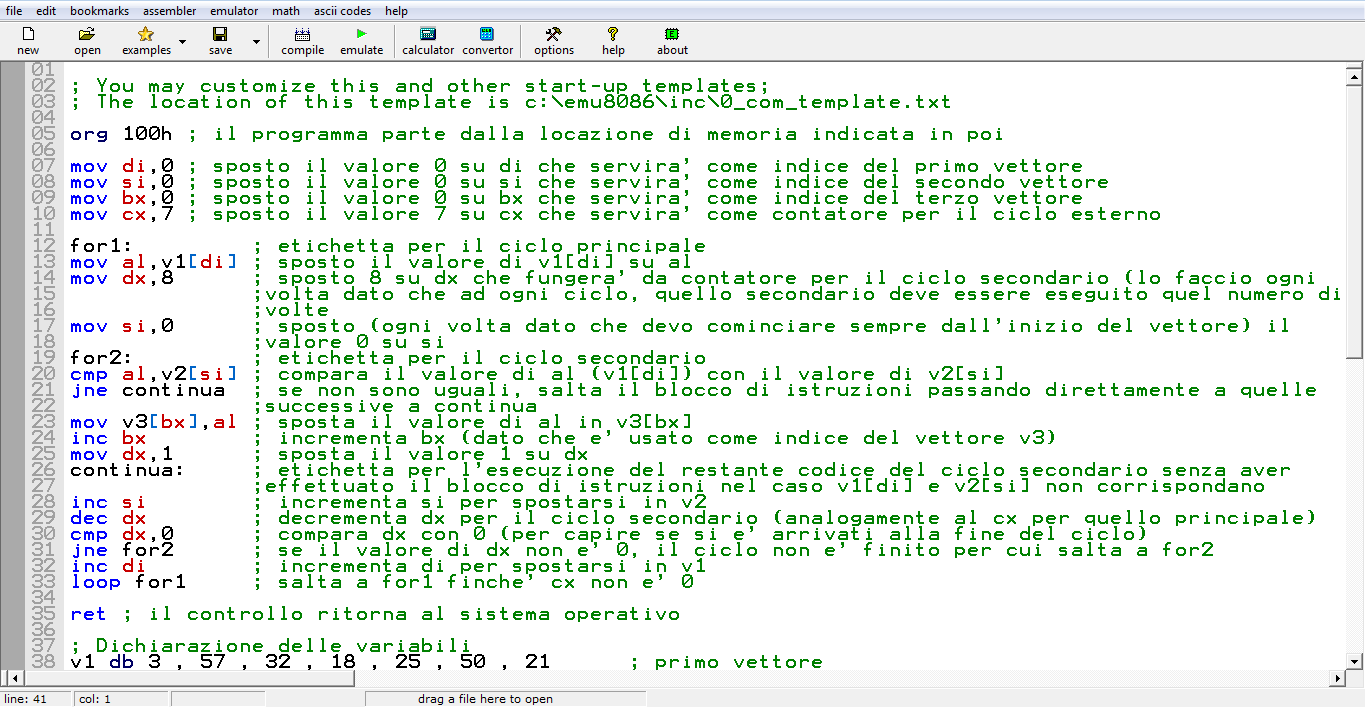
Un altro problema posto in essere durante la progettazione del programma è stato quello dello spostamento in tre array diversi: per i primi due ho potuto usare, come di consueto, i registri indice **di** e **si**, per l’ultimo, invece, sono ricorso ad un altro registro (**bx**, che era l’unico “libero” dato che gli altri erano già adibiti ad altro).

Anzitutto, ho dichiarato le variabili.

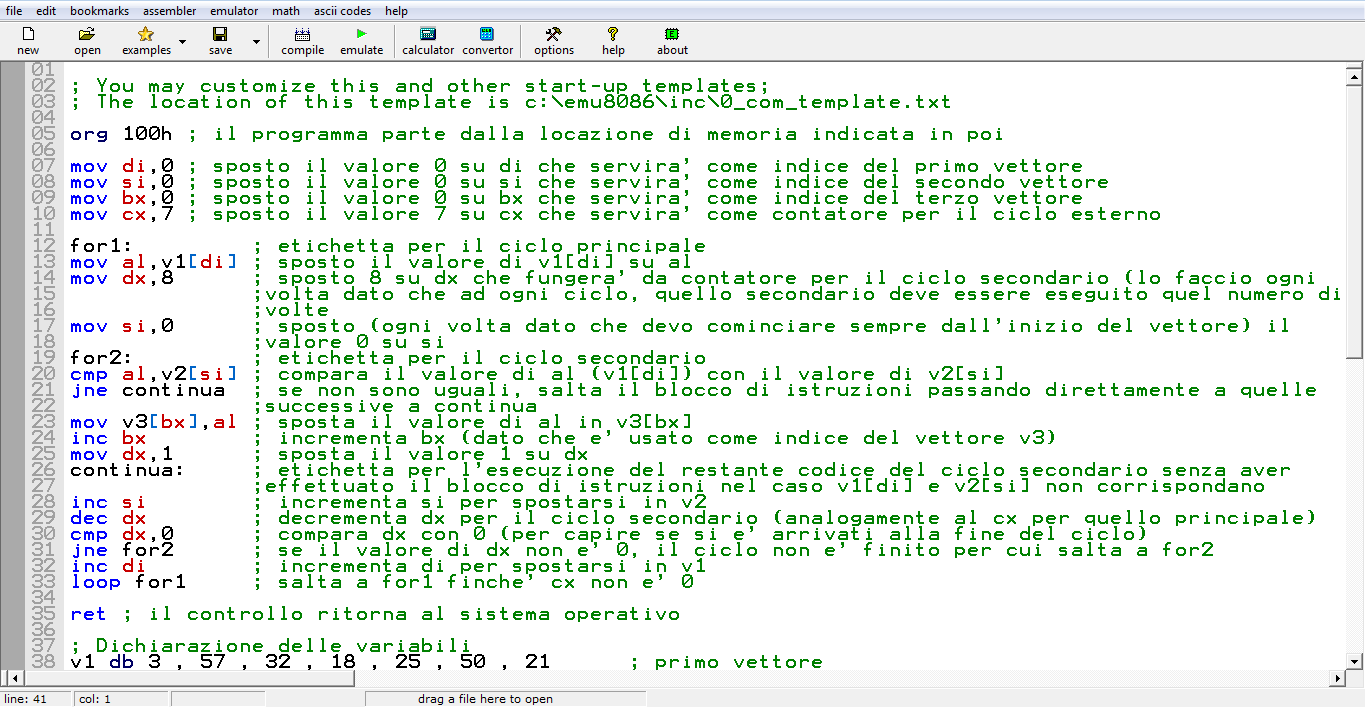


Il primo passaggio (dopo l’**org 100h** ovviamente) è stato quello di “*pulire*” i registri, atti allo spostamento nei relativi vettori, spostandoci il valore **0** (anche perché negli array si parte dalla posizione 0).

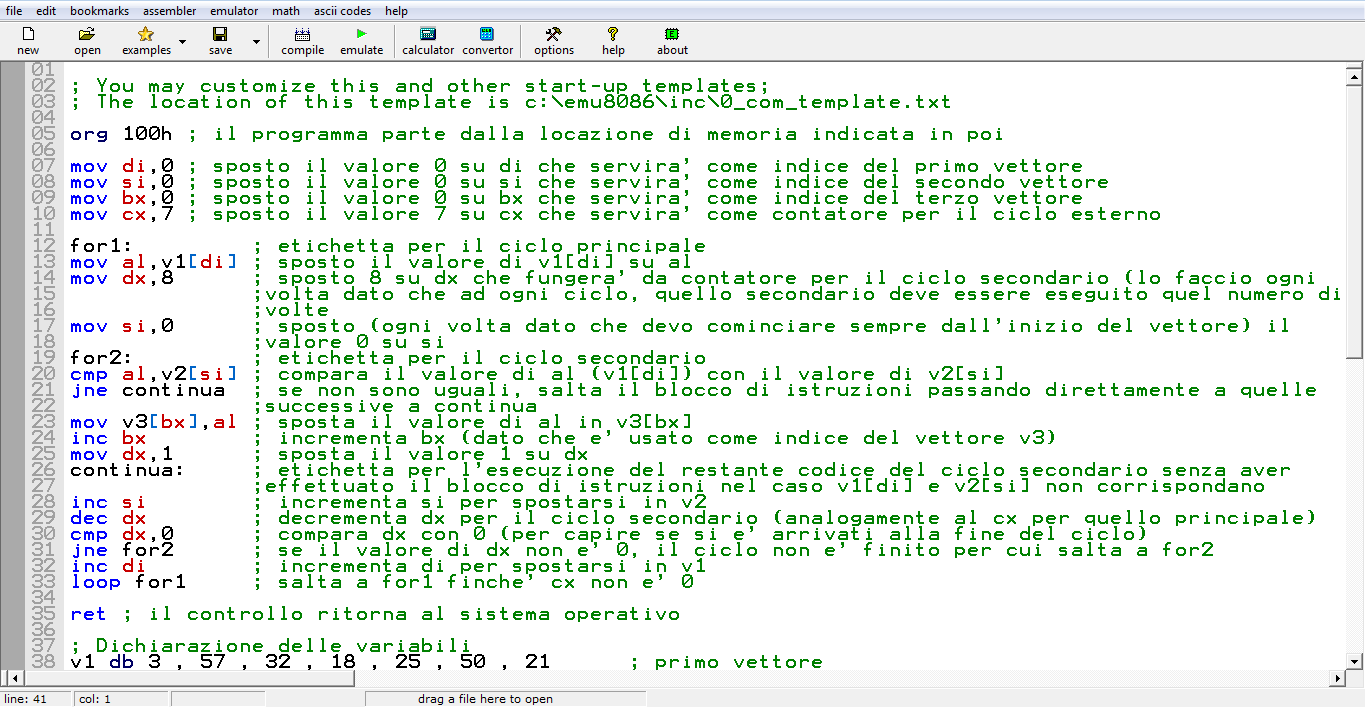
Ho spostato poi il valore 7 nel registro cx poiché il ciclo principale si dovrà ripetere un numero di volte pari al numero degli elementi contenuti in v1 (7, appunto).



Inizio il ciclo principale con l’etichetta “*for1:*”. Le istruzioni da eseguire in questo ciclo (prima dell’esecuzione di quello interno) sono lo spostamento del valore contenuto nel vettore v1 alla posizione indicata dal registro di nel registro al, e le istruzioni che permetteranno la corretta esecuzione del ciclo interno. Per eseguire quest’ultimo ho utilizzato il registro **dx** come contatore (non potendo usare **cx**), e ho spostato al suo interno 8 (dato che il numero di elementi contenuti in v2 sono 8). Ho spostato il valore 0 in **si** visto che ogni volta che si rieseguirà il ciclo più interno, bisognerà controllare i valori di v2 partendo dal primo, contenuto appunto alla posizione **0**).

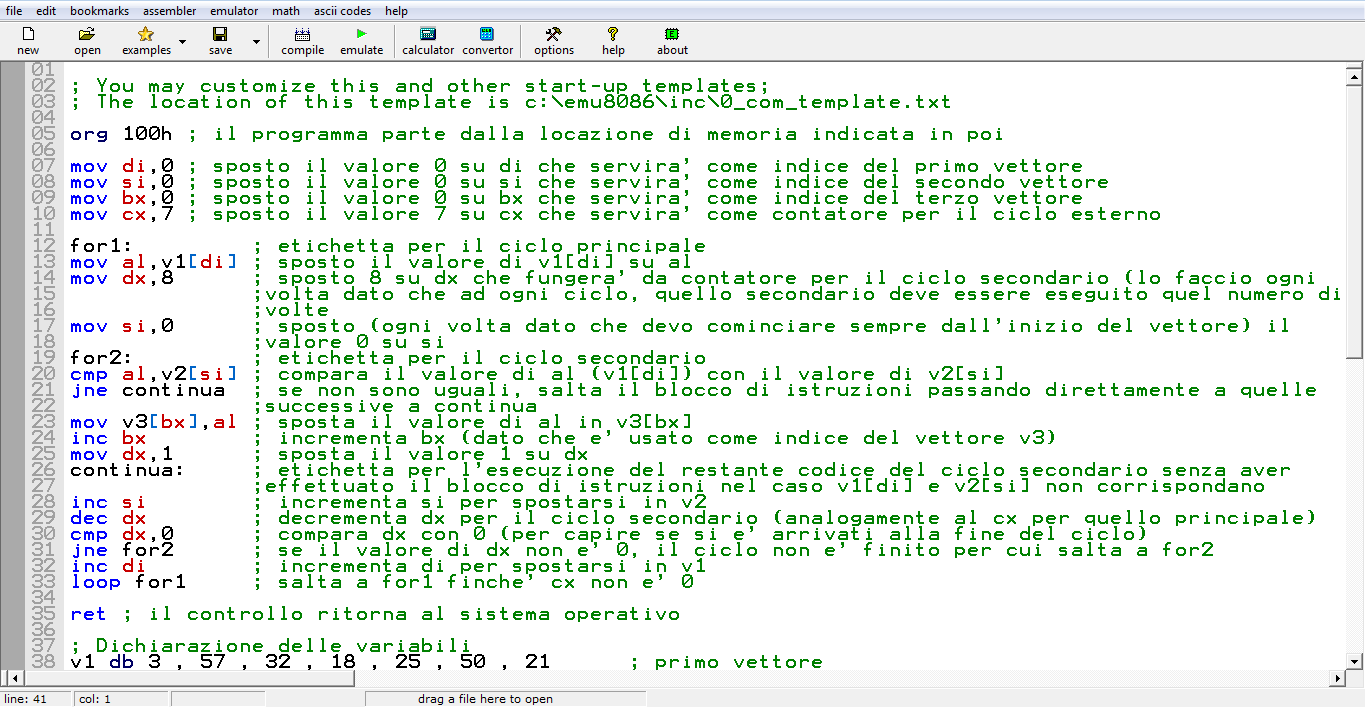


A questo punto “apro” il secondo ciclo (quello interno) con l’etichetta “*for2:*”. Dovendo compiere un confronto tra il valore del vettore 1 e quello del vettore 2, effettuo una *compare* tra **al** (in cui è contenuto il valore di **v1[di]**) e **v2[si]**. Se non sono uguali, salto direttamente a “*continua*”, altrimenti memorizzo il valore in comune ai due vettori (contenuto in **al**) nel terzo vettore in posizione bx (**v[bx]**), incremento quindi bx per spostarmi nella successiva posizione di **v3**. Poi, dato che i numeri contenuti in **v2** sono tutti diversi, ottimizzo il ciclo di controllo terminandolo una volta trovato il valore in comune, e siccome il ciclo finisce una volta che **dx** (che uso come contatore) è 0, sposto su dx 1 dato che prima di compararlo con 0 subirà un decremento.



A questo punto inserisco l’etichetta “*continua:*”, incremento si per spostarmi nella posizione successiva di v2, decremento dx, comparo quindi dx con **0**: se non è uguale (quindi il ciclo non è concluso) si ritorna ad eseguire il ciclo più interno andando a “*for2*” se, invece, dx è 0 vuol dire che il ciclo è terminato, per cui passo al valore successivo di v1 incrementando **di**, eseguo quindi il ciclo più esterno tornando a “*for1:*” finché **cx** non è **0** con **loop for1**.

Infine, con l’istruzione **ret**, il controllo ritorna al sistema operativo.



In conclusione ho controllato il corretto funzionamento del programma verificando che i valori comuni di **v1** e **v2** fossero inseriti in **v3**.

(Immagine di riferimento nella pagina seguente).

