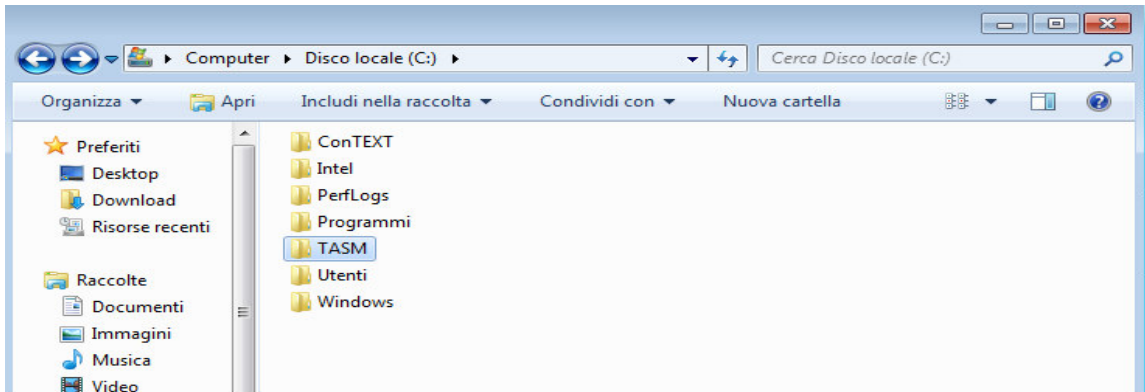


# Assembly con il TASM

Operazioni svolte in classe:

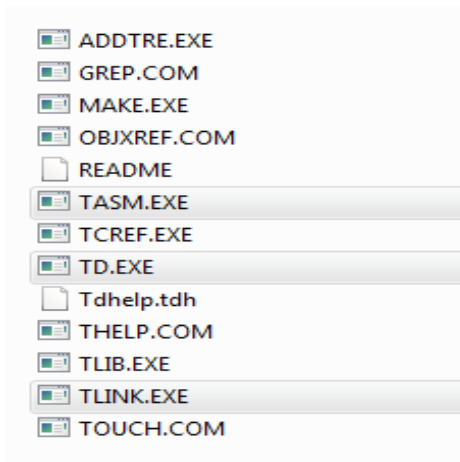
- 1) Creare su C:\ la cartella **TASM**



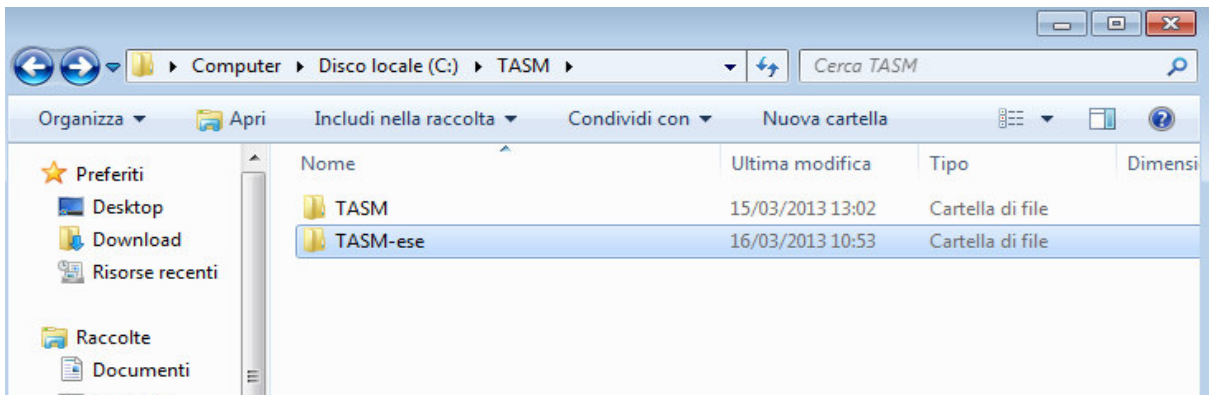
- 2) Al suo interno creare due sottocartelle:

**TASM** che conterrà l'assemblatore (TASM.EXE), il linker (TLINK.EXE) e il debugger (TD.EXE)

[ In questa cartella va estratto il file TASM.zip che deve contenere almeno il tre file .exe evidenziati qui sotto]



**TASM-ese** dove salveremo i nostri programmi sorgente (.asm), i programmi oggetto (.obj) e gli eseguibili (.exe)



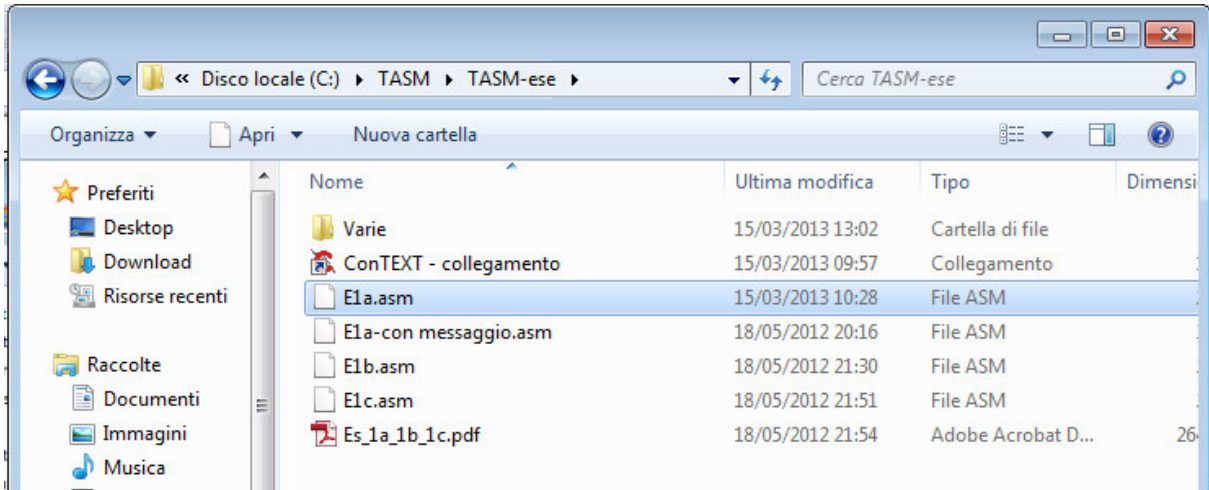
- 3) Creare il seguente programma **E1a.asm** con un editor multilinguaggio (ad esempio ConTEXT, impostando il linguaggio Assembler) o anche con Blocco note (purché poi si salvi il file con estensione **.asm**) e salvarlo nella cartella **TASM-ese**

```

ConTEXT - [K:\x-TASM da copiare su C:\TASM-ese\E1a.asm *]
File Modifica Visualizza Formato Progetti Strumenti Opzioni Finestra ?
E1a.asm *
5 10 15 20 25 30 35 40 45 50 55 60 65 70 75 80 85 90 95 100 105 110
1 ; Ciclo per leggere un carattere da tastiera senza eco e visualizzarlo sulla riga successiva
2 ; L'inserimento dei caratteri termina quando viene inserito il carattere '.' (punto)
3 ; L'inserimento della sequenza di caratteri deve essere preceduto dal messaggio "Inserisci carattere (. =fine)"
4 ;=====
5 ; E1a.asm
6 ;=====
7 .MODEL SMALL
8
9 .DATA
10     car    DB    ?
11     mess   DB    "Inserisci caratteri (. =fine)",0Ah,0Dh,"$" ; Le stringhe da visualizzare
12                                           ; devono terminare con il simbolo $
13
14 .STACK                                ;direttiva che dichiara il segmento STACK
15
16 .CODE                                ;direttiva che dichiara l'inizio del segmento Codice
17
18 Inizio:
19
20     MOV    AX,@DATA ;Inizializzazione del registro del segmento dati
21     MOV    DS,AX
22
23     ;----- programma
24     LEA    DX,mess   ; Visualizza messaggio di inserimento caratteri
25     MOV    AH,09
26     INT    21h
27 ciclo:
28     MOV    AH,07     ; Acquisisce un carattere senza eco
29     INT    21h      ; e il carattere letto viene posto dal sistema in AL
30     MOV    car,AL   ; Salvo il carattere in memoria nella variabile car
31
32     MOV    DL,car    ; Visualizzo il carattere letto
33     MOV    AH,02
34     INT    21h
35
36     CMP    car, "."  ; Confronta se il carattere è '.' (punto)
37     JNE    ciclo    ; Finché il carattere non è '.' ne acquisisco un altro
38                       ; tornando su all'etichetta ciclo:
39
40     MOV    AL,0      ;ritorno al sistema operativo
41     MOV    AH,4Ch
42     INT    21h
43     END    Inizio
44
45

```

4) Nella cartella **TASM-ese** abbiamo, ora, **E1a.asm**



5) Avviare il **Prompt del DOS**

Con il comando: **CD \** posizionarsi su **C:\>** e digitare poi i seguenti comandi in grassetto:

```
C:\>cd TASM
```

```
C:\TASM>cd TASM-ese
```

```
C:\TASM\TASM-ese>dir per controllare i nomi dei file presenti nella cartella
```

```
Il volume nell'unità C non ha etichetta.  
Numero di serie del volume: D082-31A2
```

```
Directory di C:\TASM\TASM-ese
```

```
19/03/2013  22:15    <DIR>          .  
19/03/2013  22:15    <DIR>          ..  
15/03/2013  09:57                875 ConTEXT - collegamento.lnk  
18/05/2012  19:16            2.127 E1a-con messaggio.asm  
15/03/2013  10:28            1.709 E1a.asm  
18/05/2012  20:30            1.741 E1b.asm  
18/05/2012  20:51            1.441 E1c.asm  
18/05/2012  20:54           270.117 Es_1a_1b_1c.pdf  
19/03/2013  22:15    <DIR>          Varie  
                6 File           278.010 byte  
                3 Directory  143.737.749.504 byte disponibili
```

6) Il programma **E1a.asm** vediamo che è presente nell'elenco anche nella finestra DOS.

7) Ora vogliamo:

**assemblare** il programma sorgente **E1a.asm** che si trova nella sottocartella TASM-exe, attivando l'assemblatore che si trova nella sottocartella TASM

**e salvare** il programma oggetto **E1a.obj** prodotto dall'assemblatore nella cartella in cui ci troviamo, TASM-exe.

Per fare questo dovremmo scrivere il seguente comando in grassetto ogni volta che vogliamo assemblare, specificando il nome del sorgente:

```
C:\TASM\TASM-ese>. \TASM\TASM E1a
```

Per semplificare possiamo scrivere questo comando che ci consente di specificare il percorso (path) sottinteso in cui cercare ciò che non trova nell'attuale cartella:

```
C:\TASM\TASM-ese>set path=c:\TASM\TASM
```

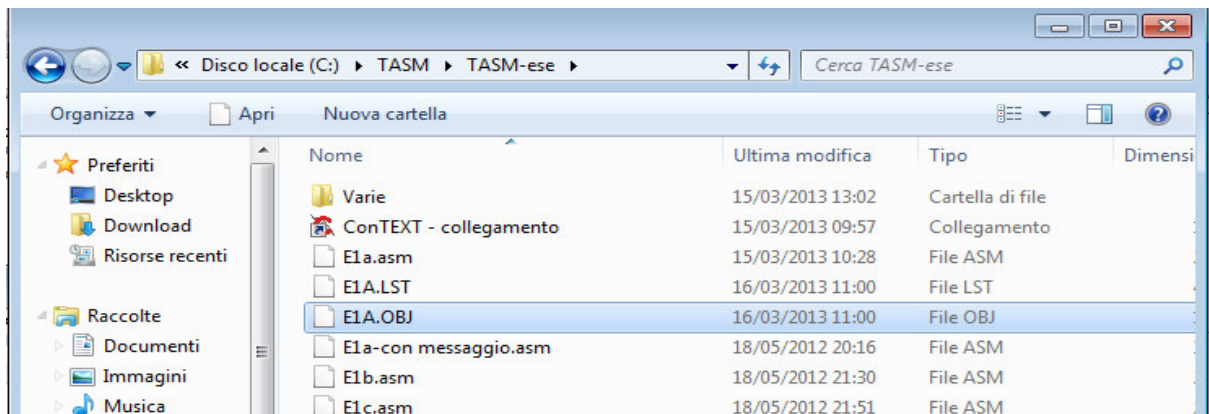
Dopodiché possiamo più semplicemente scrivere:

```
C:\TASM\TASM-ese>TASM E1a
```

E la risposta sarà del tipo:

```
Turbo Assembler Version 2.02 Copyright (c) 1988, 1990 Borland International
Assembling file:  e1a.ASM
Error messages:   None
Warning messages: None
Passes:          1
Remaining memory: 470k
```

- 8) Se non si verificano errori viene prodotto, come già detto il *file oggetto E1a.obj*, come mostra la figura sotto ( *se invece vengono rilevati degli errori si deve ripartire dal passo 3) per correggerli* ):

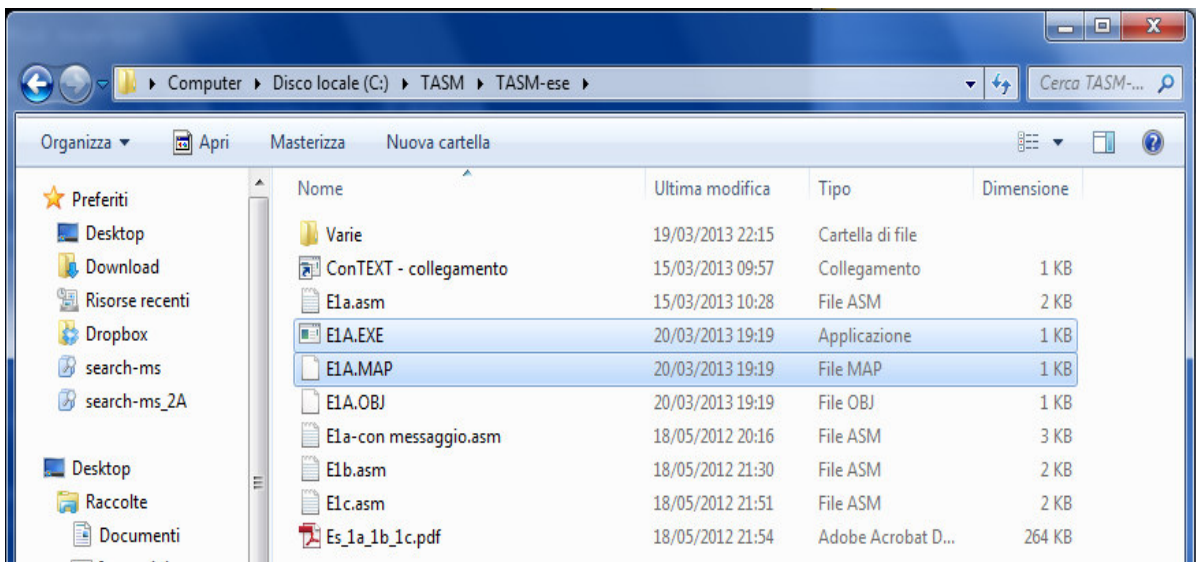


- 9) Il comando successivo serve per creare il file eseguibile attivando il Linker. Basta scrivere:

```
C:\TASM\TASM-ese>TLINK E1a
```

```
Turbo Link Version 3.01 Copyright (c) 1987, 1990 Borland International
```

E se, anche in questo caso non si verificano errori, viene prodotto il file **E1a.exe**. Di solito viene creato anche il file E1a.map.



10) Per eseguire il programma basta scrivere il nome del programma **E1a** e dare invio:

```
C:\TASM\TASM-ese>E1a
```

```
Inserisci caratteri (. =fine)
ue eu eu uieiuq iuqei qiuweoquie oqiwue qwe qwe.
```

11) Nel caso in cui il programma non contenga istruzioni di acquisizione da tastiera o di visualizzazione sul monitor, per controllarne il corretto funzionamento occorre aprire la finestra di TurboDebug (programma TD):

```
C:\TASM\TASM-ese>TD E1a
```

