

```

    . 5 . 10 . 15 . 20 . 25 . 30 . 35 . 40 . 45 . 50 . 55 . 60 . 65 . 70 . 75 . 80 .
1  ;Programma
2  ; Chiede un numero di una cifra e stabilisce se è pari o dispari
3  ; e se è superiore a 3 (usando solo operatori di bitwise AND )
4  ;=====
5  ; ES4_F2.asm
6  ;=====
7  .MODEL SMALL
8  .DATA
9      cifra    DB  ?
10     mess     DB  0Dh,0Ah,"Inserisci una cifra: $"
11     messdisp DB  0Dh,0Ah,"La cifra inserita è dispari ",0Dh,0Ah,"$"
12     messpari DB  0Dh,0Ah,"La cifra inserita è pari ",0Dh,0Ah,"$"
13     messsup3 DB  0Dh,0Ah,"La cifra inserita è superiore a 3 ",0Dh,0Ah,"$"
14     messnosup3 DB 0Dh,0Ah,"La cifra inserita non è superiore a 3 ",0Dh,0Ah,"$"
15 .STACK      ;direttiva che dichiara il segmento STACK
16
17 .CODE      ;direttiva che dichiara l'inizio del segmento Codice
18
19 Inizio:
20     MOV AX,@DATA ;Inizializzazione del registro del segmento dati
21     MOV DS,AX
22     ;----- programma
23 ciclo: LEA DX,mess
24         MOV AH,09
25         INT 21h
26         MOV AH,01 ; acquisisce cifra
27         INT 21h
28         CMP AL,30h ; Confronto se è una cifra da 0 a 9
29         JB  ciclo ; se non lo è si richiede
30         CMP AL,39h
31         JA  ciclo
32         MOV cifra,AL
33
34         AND AL,1 ; controllo se è dispari 00000001b
35         JZ  pari
36         LEA DX,messdisp ; dispari
37         MOV AH,09
38         INT 21h
39         JMP fine1
40 pari:   LEA DX,messpari ; pari
41         MOV AH,09
42         INT 21h
43 fine1:
44         AND cifra,0Ch ; controllo se superiore a 3 00001100b
45         JZ  no_sup
46         LEA DX,messsup3 ; superiore
47         MOV AH,09
48         INT 21h
49         JMP fine2
50 no_sup: LEA DX,messnosup3
51         MOV AH,09
52         INT 21h
53 fine2:
54         MOV AL,0 ;ritorno al sistema operativo
55         MOV AH,4Ch
56         INT 21h
57     END Inizio

```