

```

1 |; Chiede due caratteri da tastiera e li visualizza
2 |; in ordine alfabetico sulla riga successiva
3 |;=====
4 |; p302_ES2.asm
5 |;=====
6 |.MODEL SMALL
7 |.DATA
8 |    Car1 DB ?
9 |    Car2 DB ?
10 |    mess DB "Inserisci un carattere $"
11 |.STACK
12 |
13 |.CODE
14 |Inizio:
15 |    MOV AX,@DATA ;Inizializzazione del registro del segmento dati
16 |    MOV DS,AX
17 |
18 |;----- programma
19 |    LEA DX,mess ; ---- Visualizza messaggio di richiesta inserimento 1° carattere
20 |    MOV AH,09
21 |    INT 21h
22 |
23 |    MOV AH,01 ; Acquisisce primo carattere con eco
24 |    INT 21h
25 |    MOV car1,AL ; e lo pone in car1
26 |
27 |    MOV DL,0Ah ; Vado a riga nuova
28 |    MOV AH,02
29 |    INT 21h
30 |    MOV DL,0Dh ; e mi posiziono a inizio riga
31 |    MOV AH,02
32 |    INT 21h
33 |
34 |    LEA DX,mess ; ---- Visualizza messaggio di richiesta inserimento 2° carattere
35 |    MOV AH,09
36 |    INT 21h
37 |
38 |    MOV AH,01 ; Acquisisce secondo carattere con eco
39 |    INT 21h
40 |    MOV car2,AL ; e lo pone in car2
41 |
42 |    MOV DL,0Ah ; Vado a riga nuova
43 |    MOV AH,02
44 |    INT 21h
45 |    MOV DL,0Dh ; e mi posiziono a inizio riga
46 |    MOV AH,02
47 |    INT 21h
48 |
49 |    MOV AL,car2 ; sposta in AL il 2° carattere
50 |    CMP AL,car1 ; Contronta con il 1°
51 |    JA vis_car1 ; -- Se il 2° > 1° allora va a vis_car1
52 |                ; altrimenti continuo con l'istruzione successiva
53 |    MOV DL,car2 ; ---- Visualizza prima il 2° e poi il 1°
54 |    MOV AH,02
55 |    INT 21h
56 |    MOV DL,car1
57 |    MOV AH,02
58 |    INT 21h
59 |    JMP Fine_se ; salto incondizionato a Fine_se
60 |Vis_car1:
61 |    MOV DL,car1 ; ---- Visualizza prima il 1° e poi il 2°
62 |    MOV AH,02
63 |    INT 21h
64 |    MOV DL,car2
65 |    MOV AH,02
66 |    INT 21h
67 |Fine_se:
68 |                ;ritorno al sistema operativo
69 |    MOV AH,4Ch
70 |    INT 21h
71 |    END Inizio

```

```

1 ; Presenta un messaggio di richiesta inserimento di una cifra da tastiera
2 ; controlla che sia una cifra da 0 a 9, altrimenti la richiede,
3 ; e stabilisce se è pari o dispari visualizzando un opportuno messaggio
4 ;=====
5 ; p302_ES3.asm
6 ;=====
7 .MODEL SMALL
8 .DATA
9 Mess_ins      DB  0Dh,0Ah,"Inserisci una cifra: $" ; 0Dh e 0Ah per visualizzare
10 ; il messaggio a riga nuova
11 Mess_pari     DB  "La cifra e' pari $"
12 Mess_dispari DB  "La cifra e' dispari $"
13 Cifra        DB  ?
14 .STACK
15
16 .CODE
17 Inizio:
18     MOV  AX,@DATA ;Inizializzazione del registro del segmento dati
19     MOV  DS,AX
20
21 ;----- programma
22 Richiedi:
23     LEA  DX,mess_ins ; Visualizza messaggio di richiesta inserimento
24     MOV  AH,09
25     INT  21h
26
27     MOV  AH,01 ; Lettura da tastiera della cifra con eco
28     INT  21h
29     CMP  AL,'0' ; Confronta con il carattere ASCII 0
30     JB  Richiedi ; Finché è minore di '0' ---> non è una cifra
31 ; e torna su a richiederla
32     CMP  AL,'9' ; Confronta con il carattere ASCII 9
33     JA  Richiedi ; Finché è maggiore di '9' ---> non è una cifra
34 ; e torna su a richiederla
35 ; ----- E' una cifra
36     SUB  AL,30h ; La trasforma da carattere ASCII in numero
37     MOV  cifra, AL
38
39     MOV  DL,0Ah ; Va a riga nuova
40     MOV  AH,02
41     INT  21h
42     MOV  DL,0Dh ; e si posiziona a inizio riga
43     MOV  AH,02
44     INT  21h
45
46     MOV  AH,0 ; Prepara AX per dividere per 2
47     MOV  AL,cifra
48     MOV  BL,2 ; Porta in BL il divisore 2
49     DIV  BL ; Divide AX per BL e dopo la divisione
50 ; in AL c'è il quoziente e in AH il resto
51     CMP  AH,0 ; Confronta il resto con 0
52     JE  Pari ; -- Se il resto = 0 va all'etichetta Pari
53 ; altrimenti prosegue con l'istruzione successiva
54     LEA  DX,mess_dispari ;-- Visualizza messaggio: numero dispari
55     MOV  AH,09
56     INT  21h
57     JMP  Fine_se ; salto incondizionato a Fine_se
58 Pari:
59     LEA  DX,mess_pari ;-- Visualizza messaggio: numero pari
60     MOV  AH,09
61     INT  21h
62 Fine_se:
63     MOV  AH,4Ch
64     INT  21h
65     END Inizio

```

```

1 ; Presenta un messaggio di inserimento di una cifra da 1 a 9;
2 ; se il carattere acquisito non è una cifra viene chiesta nuovamente;
3 ; poi visualizza un quadrato fatto da asterischi di lato pari alla cifra inserita
4 ;=====
5 ; p302_ES4.asm
6 ;=====
7 .MODEL SMALL
8 .DATA
9   Mess_ins  DB  0Dh,0Ah,"Inserisci una cifra da 1 a 9: $" ; 0Dh e 0Ah per visualizzare
10 ; il messaggio a riga nuova
11   Cifra     DB  ?
12
13 .STACK
14
15 .CODE
16 Inizio:
17   MOV  AX,@DATA ;Inizializzazione del registro del segmento dati
18   MOV  DS,AX
19
20 ;----- programma
21 Richiedi:
22   LEA  DX,mess_ins ; Visualizza messaggio per inserimento
23   MOV  AH,09
24   INT  21h
25
26   MOV  AH,01 ; lettura da tastiera della cifra con eco
27   INT  21h ; il carattere ricevuto si trova in AL
28
29   CMP  AL,'1' ; Confronta con il carattere ASCII 0
30   JB  Richiedi ; Finché è minore di '1' ---> non è una cifra da 1 a 9
31 ; e torna su a richiederla
32   CMP  AL,'9' ; Confronta con il carattere ASCII 9
33   JA  Richiedi ; Finché è maggiore di '9' ---> non è una cifra da 1 a 9
34 ; e torna su a richiederla
35 ; ----- E' una cifra
36   SUB  AL,30h ; La trasforma da carattere ASCII in numero
37   MOV  cifra,AL
38
39   MOV  BL,0 ; contatore righe
40
41 ciclo_righe:
42   CMP  BL,cifra ; Confronta BL con la Cifra Letta
43   JE  Fine_ciclo ; Quando è uguale salta a Fine_ciclo
44
45   MOV  DL,0Ah ; Va a riga nuova
46   MOV  AH,02
47   INT  21h
48   MOV  DL,0Dh ; e si posiziona a inizio riga
49   MOV  AH,02
50   INT  21h
51
52   MOV  CH,0 ; Usa CX come contatore colonne da decrementare fino a quando
53 ; non diventa 0 (registro abbinato all'istruzione LOOP)
54   MOV  CL,cifra ; imposta num. asterischi di ogni riga
55
56 Ciclo_colonne:
57   MOV  DL,'*' ; visualizza un asterisco
58   MOV  AH,02h
59   INT  21h
60   LOOP Ciclo_colonne ; LOOP equivale a DEC CX e JNZ Ciclo_colonne
61
62 ; --- Fine di una riga di "*"
63   INC  BL ; Incrementa il contatore righe
64   JMP  Ciclo_righe ; Torna su per visualizzare la prossima riga di "*"
65
66 Fine_ciclo: ;ritorno al sistema operativo
67   MOV  AH,4Ch
68   INT  21h
69   END Inizio

```