

```

1:  ;=====
2:  ;=====
3:  ; PROGRAMA                      : pe13.asm
4:  ; FUNCION                       : Programa que muestra EL USO DE LOOP.
5:  ; REALIZADO POR                 : Prof. Juan Ju?rez Fuentes
6:  ; COMPILADOR EN EL QUE SE EJECUTO: GUI Turbo Assembler Version 5.1
7:  ; FECHA                         : 20240424085427
8:  ;=====
9:  ; DECLARACION DE CONSTANTES
10: ;=====
11: CR      EQU 13      ; Constante CR (Retorno de carro)
12: LF      EQU 10      ; Constante LF (Salto de l?nea)
13: TECLADO_8 EQU 8
14: IMPRIMIR_9 EQU 9
15: FIN_4C00H EQU 4C00H
16: DOS_21H EQU 21H
17: BIOS_10H EQU 10H
18: TAB      EQU 09H
19: BELL      EQU 07H
20: ;=====
21: ; DECLARACION DEL SEGMENTO DE DATOS
22: ;=====
23: DATOS SEGMENT
24:     RENGLON DB 2
25:     COLUMNA DB 18
26:     ;-----
27:     MENSAJE DB 'PROGRAMA QUE MUESTRA EL USO DE LOOP',CR,LF,'$'
28:     LETRA   DB 'A',CR,LF,'$'
29: DATOS ENDS
30: ;=====
31: ; DECLARACION DEL SEGMENTO DE DATOS
32: ;=====
33: PILA SEGMENT STACK
34:     db 32 DUP('0')
35: PILA ENDS
36: ;=====
37: ; DECLARACION DEL SEGMENTO DE CODIGO
38: ;=====
39: CODIGO SEGMENT
40:     ;-----
41:     MAIN PROC FAR ; Inicio procedimiento MAIN
42:     ;-----
43:     ASSUME CS:CODIGO,DS:DATOS,SS:PILA
44:     ;-----
45:     CALL ENCABEZADO
46:     ;-----
47:     MOV CX,0H ; LIMPIA EL CONTADOR A CERO,D ELO CONTRARIO PODRIA NO CONTAR BIEN.
48:     MOV CL,05H
49:     CICLO:
50:     MOV AH,09H
51:     LEA DX,LETRA
52:     INT 21H
53:     LOOP CICLO
54:     ;-----
55:     CALL FIN
56:     ;-----
57:     MAIN ENDP ; Fin del procedimiento MAIN
58:     ;-----
59:     IMPRIME_TEXTO PROC NEAR
60:     MOV AH,IMPRIMIR_9 ; Funci?n DOS para escribir texto en pantalla
61:     INT DOS_21H ; Llamado a la interrupci?n del DOS
62:     RET
63:     IMPRIME_TEXTO ENDP
64:     ;-----

```

```
65:      FIN PROC NEAR
66:      MOV AX,FIN_4C00H ; Fin del programa
67:      INT DOS_21H
68:      RET
69:      FIN ENDP
70:      ;-----
71:      LIMPIA_PANTALLA PROC NEAR
72:      MOV AX,0600H ; Establecer modo texto y CLS
73:      MOV BH,1FH ; Colores de fondo (negro=0) y texto (verde=A).
74:      MOV CX,0000H ; Desde el inicio de pantalla
75:      MOV DX,484FH ; Hasta el fin de la pantalla Reng (48)72 col(79)4F
76:      INT BIOS_10H ; Interrupci?n que llama al BIOS
77:      RET
78:      LIMPIA_PANTALLA ENDP
79:      ;-----
80:      POSICIONA_CURSOR PROC NEAR
81:      MOV AH,02 ; Servicio para Fija posicion del Cursor
82:      MOV BH,00 ; Numero de pagina o pantalla
83:      MOV DH,REGLON
84:      MOV DL,COLUMN
85:      INT BIOS_10H
86:      RET
87:      POSICIONA_CURSOR ENDP
88:      ;-----
89:      ENCABEZADO PROC NEAR
90:      CALL LIMPIA_PANTALLA
91:      ;-----
92:      MOV AX,DATOS ; AX=Direcci?n del segmento de datos
93:      MOV DS,AX ; DS=AX. Indicar el segmento de datos
94:      ;-----
95:      CALL POSICIONA_CURSOR ; Llama a procedimiento par posionar cursor.
96:      ;-----
97:      LEA DX,MENSAJE ; DS:DX = direcci?n de MENSAJE
98:      CALL IMPRIME_TEXTO
99:      ;-----
100:     RET
101:     ENCABEZADO ENDP
102:     ;=====
103:     CODIGO ENDS
104:     ;=====
105:     END MAIN
106:     ;=====
107:     ;=====
108:     ;=====
```