

```

1  TITLE    INOUT10  CSCI 225
2  NAME INOUT10
3  ;DC Pankratz (All Rights Reserved)
4
5
6  COMMENT ! These procedures input and output chars, strings,
7           hex numbers from and to the keyboard and monitor
8           respectively. BIOS interrupt 10h is used for the IO.
9
10 To use these routines you must name your code segment as follows
11 CSEG SEGMENT PUBLIC
12
13 Also, define all procedures called from this file as EXTRN NEAR
14 within your code segment.
15 !
16
17
18 ;*****
19 ;                               I N O U T 1 0
20 ;*****
21
22 CSEG          SEGMENT PUBLIC
23 Assume CS:CSEG
24
25 PUBLIC putDec10, putCstring10
26
27 ; these procs are in BIOS.asm
28 EXTRN where:near, write:near, locate:near
29 EXTRN writeattr:near
30
31
32 putCstring10 PROC NEAR
33 COMMENT ! This procedure outputs a character at a time from
34           consecutive memory locations until NULL = 0.
35           IN: SI contains the starting address for the string.
36               BL contains the color attribute.
37           OUT: none
38           !
39           PUSH SI           ; Value parameter
40           PUSH AX          ; Local variable
41           PUSH DX
42           PUSH CX
43
44           Call Where         ; get cursor position
45           MOV CX, 1         ; number of chars to write
46 SOagain:
47           MOV AL,[SI]
48           CMP AL,0          ; If end of string detected,
49           JE SOdone         ; exit loop and return.
50
51           call locate
52           call writeattr
53           inc DL           ; column
54
55           INC SI           ; Get ready for next char.
56           JMP SOagain
57 SOdone:
58           POP CX
59           POP DX
60           POP AX          ; Local variables.
61           POP SI          ; Value parameter.
62           RET             ; Return to calling program.
63 putCstring10 ENDP
64
65 ;*****
66 ;*****
67
68
69 putDec10 PROC near

```

```

70  Comment!           Displays the number contained in DX in signed decimal
71                      representation.
72  IN:  DX = the number to be displayed with current attribute
73  OUT: none
74  !
75      Push          AX
76      Push          BX
77      Push          CX
78      Push          DX
79      Push          SI
80      mov  ax, 0
81      push ax          ; sentinel
82      Mov  BX, 10
83
84      mov  SI, 0
85      Cmp  DX, 0        ; check for negative
86      JGE  D0sign
87      inc  SI          ; negative flag
88      Neg  DX
89  D0sign:      Mov  AX, DX
90  DONextDigit:
91      xor  DX, DX
92      DIV  BX
93      add  dl, 30h
94      Push DX
95      Cmp  AX, 0
96      JA  DONextDigit
97
98      Call where
99      mov  CX, 1        ; number of chars to write
100     cmp  SI, 0
101     jz   D0Display
102     mov  AL, '-'
103     call write
104     inc  DL
105  D0Display:
106     Pop  AX
107     cmp  ax, 0
108     jz   D0done
109     call locate
110     Call write
111     inc  DL          ; next column
112     jmp  D0Display
113  D0done:
114     Pop  SI
115     Pop  DX
116     Pop  CX
117     Pop  BX
118     Pop  AX
119     Ret
120  putDec10      ENDP
121
122
123
124
125  CSEG          ENDS
126  END
127

```