Macintosh Mouse: Mouse/Cursor Tracking

There is a way to detect if the mouse was moved and how far it was moved. The
mouse location call provides the current coordinates of the mouse position. If
two successive calls to mouse location yield different results, then the mouse
has moved, and (given the previous and present coordinates) the distance
traveled can be calculated. If you wish to know the total distance the mouse
traveled between calls to mouse location, then you should use the mouse
odometer. The fact that the mouse moves is not considered an event, so it is up
to the programmer to determine the mouse's location and distance traveled. Also
note that the only thing that happens to a cursor as it travels around the
screen (without any mouse clicks) is that the cursor changes according the area
it is in--e.g., it becomes an I-beam over text, a cross in a spreadsheet, and
an arrow most other times. It should also be noted that these cursor changes
are under program control and are not automatic.

Here is a small program that uses some of the cursor/mouse tracking routines.
These routines automatically keep the cursor on the screen--i.e., if the cursor
is already at the left edge of the screen and you move the mouse further left,
the cursor stays on the screen. This program is similar to the Hendrix program
on the MacMaster 2 disk in that it makes noise.

```
PROGRAM VOL;
USES
{$U-}
     {$U QD/QuickDraw }  QuickDraw,
     {$U QD/QDSupport }  QDSupport,
     {$U QD/Hardware }  Hardware;
{$U+}

VAR heapBuf:    ARRAY[0..10000] OF INTEGER;{must keep array under 32K
byte limit}
    ABS,ORD:PIXELS;
    event:             KeyEvent;

FUNCTION HeapFull(hz: QDPtr; bytesNeeded: INTEGER): INTEGER;
{ This function will be called if the heapZone runs out of space }
BEGIN
  WRITELN('The heap is full.  The program must now terminate! ');
  Halt;
END;


BEGIN {MAIN PROGRAM}
```

```
   {----- Initialization - Generic to all applications using QuickDraw
-----}
   QDInit(@heapBuf, @heapBuf[10000], @HeapFull); { Must do this once at
beginning }

   INITCURSOR;
    SHOWCURSOR;
    REPEAT
       MOUSELOCATION(ABS,ORD);
       SETVOLUME((ABS DIV 120)+1);
       NOISE(ORD *(8000 DIV 360)+80 );
    UNTIL KEYBDEVENT(FALSE,FALSE,EVENT);
    HIDECURSOR;
    SILENCE
END.
```
Apple Technical Communications

Keywords: <None>

==

This information is from the Apple Technical Information Library.

19960215 11:05:19.00

Tech Info Library Article Number: 799