

```

*****
*                               N O K E Y P                               *
*****
* Task       : Demonstrates clearing the keyboard buffer.                *
*             This is useful for protecting the user from                *
*             accidental keystrokes during an important                   *
*             command (e.g., deleting files).                            *
*****
* Author      : Michael Tischer                                          *
* Developed on : 01/01/92                                                *
* Last update  : 02/10/92                                                *
*****

program NoKeyP;

uses Crt;                                { * Add CRT unit * }

*****
* ClearKbBuffer : Clears the contents of the keyboard buffer.            *
* Input       : None                                                    *
* Output      : None                                                    *
*****

procedure ClearKbBuffer;

begin
    inline( $fa );                    { CLI: Disable hardware interrupts }
    memw[$40:$1A] := memw[$40:$1C];   { No more characters in buffer }
    inline( $fb );                    { STI: Enable hardware interrupts }
end;

*****
*                               M A I N   P R O G R A M                               *
*****

var i,                                { Loop counter }
    ccount : integer;                 { Number of character in keyboard buffer }
    kch    : char;                   { Get keys }

begin
    clrscr;
    writeln( 'NOKEYP - (c) 1992 by Michael Tischer' );
    writeln;
    writeln( 'Keyboard buffer purged when counter reaches 0.' );
    writeln;

    ClearKbBuffer;                    { Clear the buffer }

    for i := 10 downto 0 do           { Give user time to }
    begin                             { press some keys }
        writeln( i:5 );
        delay( 750 );
    end;

    {-- Display characters still in keyboard buffer -----}

    ccount := 0;                      { No more characters }
    writeln;
    writeln;
    writeln( 'Characters in keyboard buffer : ' );

    while KeyPressed do               { Any more characters in keyboard buffer? }
    begin                             { Yes --> Read and display }
        kch := ReadKey;
        write( ' ', ord(kch):5 );     { Display code only first }
        if ord(kch) > 32 then         { Code > 32? }
            write( '(', kch, ')' );   { Yes --> Display character as well }
        writeln;
        inc( ccount );               { More than one character found }
    end;
    if ccount = 0 then                { Out of characters? }
        writeln( '(None)' );         { Done }
    writeln;
end.

```