

C listing: WINDAC.C

```

/*****
/*
/*-----
/* Task          : Determines whether Windows is active, and if so,
/*                the operating mode.
/*-----
/* Author         : Michael Tischer
/* Developed on   : 08/22/91
/* Last update    : 04/07/95
/*-----
/* Memory model  : SMALL
*****/

/*== Add include files =====*/

#include <dos.h>
#include <stdio.h>

/*== Constants =====*/

#define MULTIPLEX    0x2F /* Interrupt number of multiplex interrupt */

#define NO_WIN       0x00 /* Windows not active */
#define W_386_X      0x01 /* Windows/386 V2.x running */
#define W_REAL       0x81 /* Windows running im real mode */
#define W_STANDARD   0x82 /* Windows running im standard mode */
#define W_ENHANCED   0x83 /* Windows running im extended mode */

/*****
/* WINDOWS : Determines whether Windows is active
/* Input   : MVERSION = Pointer to INT variable to which the main

```

```

/*                      version number should be passed          */
/*          SVERSION = Pointer to INT variable to which the sub   */
/*                      version number should be passed          */
/* Output   : Windows status, from constants NO_WIN, W_386_X,    */
/*           W_STANDARD, W_STANDARD or W_ENHANCED                */
/* Info     : Version number can only be passed and returned when */
/*           Windows 3.x is operating in enhanced mode          */
/******

```

```

int windows( int *MVersion, int *SVersion )
{
    union REGS regs;                /* Registers for interrupt call */
    struct SREGS sregs;             /* Segment register for interrupt call */

    *MVersion = 0;                  /* Initialize version number */
    *SVersion = 0;

    /*-- Windows x.y in extended mode -----*/

    regs.x.ax = 0x1600;             /* Installation check for Windows */
    segread( &sregs );              /* Read segment register */
    int86x( MULTIPLEX, &regs, &regs );

    switch ( regs.h.al )
    {
        case 0x01:
        case 0xFF:  *MVersion = 2;                /* Main version */
                   *SVersion = 0;                /* Sub version unknown */
                   return W_386_X;              /* Windows/386 Version 2.x */

        case 0x00:
        case 0x80:  regs.x.ax = 0x4680;          /* Real mode and standard mode */
    }
}

```

```

int86x( MULTIPLEX, &regs, &regs, &sregs );
if ( regs.h.al == 0x80 )
    return NO_WIN;                /* Windows not running */
else
{
    /*-- Windows in real mode or standard mode -----*/

    regs.x.ax = 0x1605;           /* Emulate initialization */
    regs.x.bx = regs.x.si = regs.x.cx =
        sregs.es = sregs.ds = 0x0000;
    regs.x.dx = 0x0001;
    int86x( MULTIPLEX, &regs, &regs, &sregs );
    if ( regs.x.cx == 0x0000 )
    {
        /*-- Windows in real mode -----*/

        regs.x.ax = 0x1606;
        int86x( MULTIPLEX, &regs, &regs, &sregs );
        return W_REAL;
    }
    else
        return W_STANDARD;
}

/*-- Windows in extended mode, AX contains version number -----*/

default: *MVersion = regs.h.al;    /* Display Windows version */
        *SVersion = regs.h.ah;
        return W_ENHANCED;        /* Windows in extended mode */
}
}

```

```

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

int main( void )
{
    int WindowsActive,                                /* Windows mode */
        MVer,                                         /* Main version of Windows */
        SVer;                                         /* Sub version of Windows */

    printf(" WINDAC - (c) 1991 by Michael Tischer \n\n" );
    WindowsActive = windows( &MVer, &SVer );
    switch ( WindowsActive )
    {
        case NO_WIN:      printf( "Windows not active\n" );
                           break;
        case W_REAL:      printf( "Windows active in real mode\n" );
                           break;
        case W_STANDARD:  printf( "Windows active in standard mode\n" );
                           break;
        case W_386_X:     printf( "Windows/386 V 2.x active" );
                           break;
        case W_ENHANCED:  printf( "Windows V %d.%d active in %s\n",
                                MVer, SVer, "extended mode" );
                           break;
    }
    return( WindowsActive );
}

```