

```

// Vahe Karamian - www.karamian.com
// Filename: cpw.c

// Basic cp file copy program WIN32 implementation

// Usage: cpw file1 file2 - copy file1 to file2

/*
 windows.h is allways included and contains all Win32 function
 definitions and data types.
*/
#include <windows.h>
#include <stdio.h>

#define BUF_SIZE 256

int main( int argc, LPTSTR argv[] )
{
    /*
     All WIN32 objects are identified by variables of type HANDLE, and a
     single generic CloseHandle function applies to most objects.
    */
    HANDLE hIn, hOut;
    DWORD nIn, nOut;

    CHAR Buffer[ BUF_SIZE ];

    if( argc != 3 )
    {
        printf( "Usage: cpw file1 file2\n" );
        return( 1 );
    }

    hIn = CreateFile( argv[1], GENERIC_READ, 0, NULL, OPEN_EXISTING, 0, NULL );
    if( hIn == INVALID_HANDLE_VALUE )
    {
        printf( "Cannot open input file. Error: %x\n", GetLastError( ) );
        return( 2 );
    }

    hOut = CreateFile( argv[2], GENERIC_WRITE, 0, NULL, CREATE_ALWAYS,
        FILE_ATTRIBUTE_NORMAL, NULL );
    if( hOut == INVALID_HANDLE_VALUE )
    {
        printf( "Cannot open output file. Error: %x\n", GetLastError( ) );
        return( 3 );
    }

    while( ReadFile( hIn, Buffer, BUF_SIZE, &nIn, NULL ) && nIn > 0 )
    {
        WriteFile( hOut, Buffer, nIn, &nOut, NULL );
        if( nIn != nOut )
        {
            printf( "Fatal write error: %x\n", GetLastError( ) );
            return( 4 );
        }
    }

    /*
     It is recommended that all open handles be closed when they are no
     longer required so as to free resources. Nonetheless, the handles
     will be closed automatically when a process exits.
    */
    CloseHandle( hIn );
    CloseHandle( hOut );

    return( 0 );
}

```