# Exercises for Chapter 22– Unsafe Code and Interop

## Exercise 1

***Calling a DLL Function Exercise***

The GoodOldCFunctions DLL contains the following C function:

extern "C" void StripWhiteSpaceAndPunctuation(char\* text,  
 int\* nWhitespace, int\* nPunctuation)

{

\*nWhitespace = 0;

\*nPunctuation = 0;

char\* p = text;

do {

if (isspace(\*text)) {

++\*nWhitespace;

}

else if (ispunct(\*text)) {

++\*nPunctuation;

}

else {

\*p++ = \*text;

}

} while (\*text++);

}

* This function receives a string and removes all whitespace and punctuation characters, returning the numbers of characters that were removed
* Your task is to write a C# program which:
  + Asks the user for a string
  + Passes the string to the above function
  + Prints the original string, the altered string and the number of characters removed

## Exercise 2

***Calling Win32 APIs***

* In this exercise you should list the titles of all currently opened windows on your screen, using Win32 API functions
* The first relevant API function is **EnumWindows**
  + The **EnumWindows** function enumerates all top-level windows on the screen, passing the handle to each window, in turn, to an application-defined callback function.

**BOOL EnumWindows(**

**WNDENUMPROC lpEnumFunc, //callback function (see EnumWindowsProc below)**

**LPARAM lParam // application-defined value (ignore for now)  
 );**

* The callback function required by **EnumWindows** is:

**BOOL CALLBACK EnumWindowsProc(**

**HWND hwnd, // handle to parent window**

**LPARAM lParam // application-defined value (ignore for now)**

**);**

* For each window found by the **EnumWindows** function, print its title to the console
* Getting a window’s title can be done using another Win32 API function: **GetWindowText**
  + The **GetWindowText** function copies the text of the specified window's title bar (if it has one) into a buffer.

int GetWindowText(

HWND hWnd, // handle to window or control

LPTSTR lpString, // text buffer

int nMaxCount // maximum number of characters to copy

);

* Define appropriate wrapper methods for the Win32 API functions
* Define the callback function according to the exercise requirements
* Provide a Main() method that activates the **EnumWindows** method wrapper to get the required results