Moisés Fernández Zárate A01197049

Programación avanzada (Gpo 2)

Windows Service

Utilicé la herramienta nssm para instalar el servicio

# Cambios:

* Cambié las dirección de donde están los archivos y las direcciones de ambos archivos.
* Le agregué punto y coma a la primera variable de tipo HANDLE.
* Le agregué el tipo HANDLE a la variable fileHandle.
* Le agregué la variable fileHandle a CloseHandle en la línea 60 porque no tenía nada de parámetros.

# Código

#include <windows.h>

#include <stdio.h>

#include <stdlib.h>

#include <conio.h>

void main() {

char dirStr[]="C:\\Users\\moy-8\\Desktop\\Github\\advanced-programming\\p15";

char dirFileStr[]="C:\\Users\\moy-8\\Desktop\\Github\\advanced-programming\\p15\\Algo.txt";

char dirFileStr2[]="C:\\Users\\moy-8\\Desktop\\Github\\advanced-programming\\p15\\Algo2.txt";

HANDLE changeNotifHandle;

HANDLE fileHandle;

char chr = 0;

DWORD waitResult;

int flag=0;

FILETIME lastWriteTime,t1,t2,t3;

DWORD lastError;

//Asegurar tiempos de facto para el archivo

lastWriteTime.dwHighDateTime=0; //Asignar los 32bits HIGH

lastWriteTime.dwLowDateTime=0; //Asignar los 32bits LOW

//Abrir un "handle" con el archivo para obtener (inicialmente) el tiempo de ultima escritura

fileHandle=CreateFile(dirFileStr,GENERIC\_READ,FILE\_SHARE\_READ,NULL,OPEN\_EXISTING,0,NULL);

//Si el

if (fileHandle!=INVALID\_HANDLE\_VALUE) {

if(GetFileTime(fileHandle,&t1,&t2,&t3))

memcpy(&lastWriteTime,&t3,sizeof(FILETIME));

CloseHandle(fileHandle);

} else {

lastError=GetLastError();

if (lastError==ERROR\_FILE\_NOT\_FOUND)

return; //Archivo no localizado, quizas lo borraron ... terminar el programa

//Otra posibilidad del error es que el archivo este abierto y LOCKED ...

// ... habrï¿½ que decidir como proceder (terminar el programa tambiï¿½n o ciclarse esperando)

// ... pero esto no lo vamos a hacer aquï¿½ simplemente tomamos valores de tiempo en ceros.

}

while (chr!=27) {

if (!flag)

changeNotifHandle=FindFirstChangeNotification(dirStr,FALSE,FILE\_NOTIFY\_CHANGE\_LAST\_WRITE);

else

FindNextChangeNotification(changeNotifHandle);

waitResult = WaitForSingleObject(changeNotifHandle,500); //Esperar hasta 500 mseg.

flag=1;

switch(waitResult) {

case WAIT\_OBJECT\_0:

printf("Cambio algun archivo\n");

Sleep(50); //Dar un tiempo para evitar el "LOCK" cuando se esta grabando el archivo

fileHandle=CreateFile(dirFileStr,GENERIC\_READ,FILE\_SHARE\_READ,NULL,OPEN\_EXISTING,0,NULL);

if (fileHandle!=INVALID\_HANDLE\_VALUE) {

if(GetFileTime(fileHandle,&t1,&t2,&t3)) {

//printf("Time: %ld, %ld\n",t3.dwHighDateTime,t3.dwLowDateTime);

if ((t3.dwHighDateTime!=lastWriteTime.dwHighDateTime)||(t3.dwLowDateTime!=lastWriteTime.dwLowDateTime)) {

printf("El archivo cambio\n");

memcpy(&lastWriteTime,&t3,sizeof(FILETIME));

//unlink(dirFileStr2);

CopyFile(dirFileStr,dirFileStr2,FALSE);

} else {

printf("El archivo NO cambio\n");

}

}

CloseHandle(fileHandle);

} else {

lastError=GetLastError();

if (lastError == ERROR\_SHARING\_VIOLATION) //ï¿½Ups!, que mala suerte el archivo cambio y esta LOCKED

printf("El archivo cambio!\n");

else if (lastError == ERROR\_FILE\_NOT\_FOUND) {

printf("Borraron el achivo ... que hacemos?\n");

return; //Por lo pronto finalizar este programa

} else

printf("INVALID\_HANDLE\_VALUE! ... %ld\n",lastError);

}

break;

case WAIT\_TIMEOUT:

printf(".");

break;

default:

printf("ERROR!!!\n");

break;

}

if (kbhit())

chr=getch();

}

FindCloseChangeNotification(changeNotifHandle);

printf("Listo\n");

}

# Pruebas



