Urządzenia Peryferyjne

Karta muzyczna

Nikita Stepanenko, 245816

Termin zajęć Środa 11:00 TP

# 1.Cel ćwiczenia

Celem wykonywanego ćwiczenie było stworzenie programu, który będzie umożliwiać:  
a) wybór pliku od odtworzenia  
b) wybór odtwarzania wybranego pliku metodą:  
 1. PlaySound   
 2. Windows Media Player (WMP)  
 3. WaveOutWrite  
 4. MCI  
 5. Direct Sound

Dla każdej metody należało umieścić dwa przyciski START i STOP oraz (jeżeli metoda umożliwia) przycisk PAUSE.

Następnym celem było rozszerzyć ten program w części Direct Sound o możliwość dodawania efektów dźwiękowych.

Ostatnim celem było rozszerzyć program o możliwość nagrywania dźwięku jedną z metod: WaveInWrite, MCI lub DirectSound

# 2.Wstęp teoretyczny

Karta muzyczna – komputerowa karta rozszerzeń umożliwiająca rejestrację, przetwarzanie i odtwarzanie dźwięku. Obecnie układy dźwiękowe wystarczające do zastosowań amatorskich są zazwyczaj wbudowywane w [płytę główną](https://pl.wikipedia.org/wiki/P%C5%82yta_g%C5%82%C3%B3wna) komputera, a nie stanowią karty rozszerzenia. Z powodów historycznych są jednak określane mianem „zintegrowana karta dźwiękowa”. Pojawiły się również zewnętrzne karty dźwiękowe podłączane do komputera przez port [USB](https://pl.wikipedia.org/wiki/USB).

# 3.Realizacja ćwiczenia

W trakcie zajęć udało się wykonać następne zadania:   
1. Napisać program, który umożliwia odtwarzanie wybranego pliku typu WAV następnymi metodami:

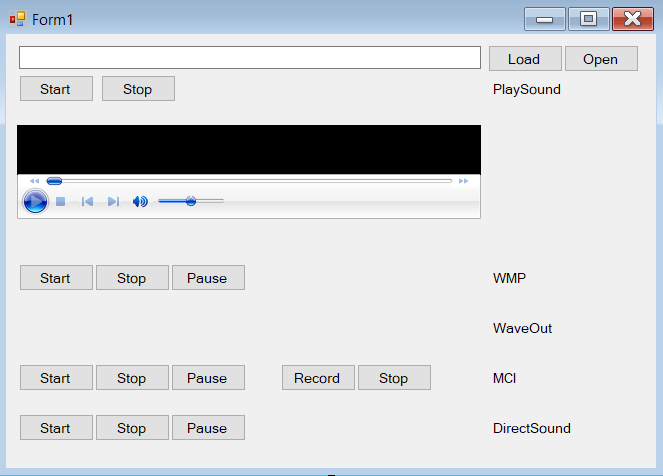
1. PlaySound
2. Windows Media Player
3. MCI
4. DirectSound.

2. Rozszerzyć ten program o możliwość nagrywania dźwięku metodą MCI.

Program został napisany z wykorzystaniem języka obiektowego C#. Po uruchomieniu programu mamy możliwość wybrać plik typu WAV na komputerze, potem mamy możliwość do odtwarzania tego pliki potrzebną nam metodą. Są przyciski START oraz STOP, które posługują dla początku odtwarzania pliku oraz kończenia odpowiednio.

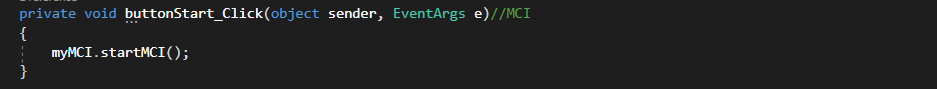
W programie realizowano 4 klasy dla obsługi plików wav: MyDirectSound, MyPlaySound, MyMCI. 1 klasa dla obsługi okna głównego oraz 1 klasa dla uproszczenia pracy z plikiem.

Dla interfejsu graficznego wykorzystywany Windows Forms.( System.Windows.Forms).



**Klasa obsługi okna:**

Ta klasa jest bardzo prosta. Po naciśnięciu odpowiedniego przycisku w oknie programu jest wywołana odpowiadająca metoda, która wywoła obiekt potrzebującej metody i wykonuje pewną funkcję. Na przykład: naciskamy „Start” dla metody MCI. Funkcja wywoła funkcję obiektu myMCI która już odtwarza plik.



**Realizacja metody PlaySound:**

**Biblioteki:**

System;

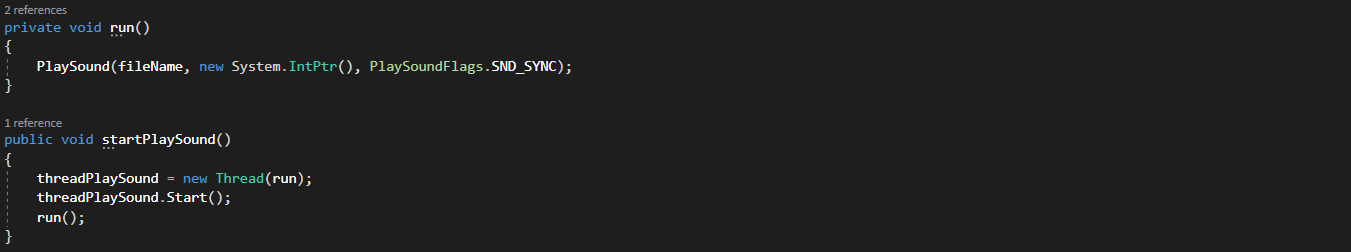
System.Threading;

System.Security.Permissions;

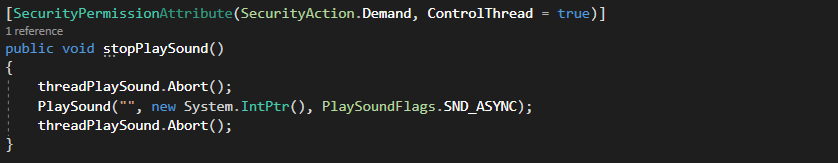
System.Runtime.InteropServices;

[System.Runtime.InteropServices.DllImport("winmm.DLL", EntryPoint = "PlaySound", SetLastError = true, CharSet = CharSet.Unicode, ThrowOnUnmappableChar = true)]

Odtwarzanie pliku. Wykorzystam dodatkowy wątek żeby główny program działał niezależnie.



Stop. Tu mam jakiś problem z zamknięciem wątku, dlatego nie działa, chociaż musi.



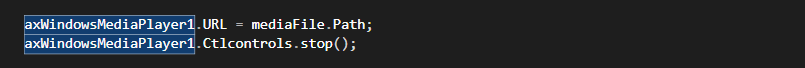
**Realizacja metody WindowsMediaPlayer:**

**Biblioteki:**

WMPLib

axWMPLib

Odtwarzanie pliku.



Start/Play.



Stop.



Pause.



**Realizacja metody MCI:**

**Biblioteki:**

System;

System.Text;

[System.Runtime.InteropServices.DllImport("winmm.dll")]

private static extern int mciSendString(string strCommand,

StringBuilder strReturn, int iReturnLength, IntPtr hwndCallback);

[System.Runtime.InteropServices.DllImport("winmm.dll")]

public static extern int mciGetErrorString(int errCode,

StringBuilder errMsg, int buflen);

Odtwarzanie pliku.

Start/Play.

Stop.



Pause.

  
  
Nagrywanie.

  
  
Stop nagrywanie.



**Realizacja metody DirectSound:**

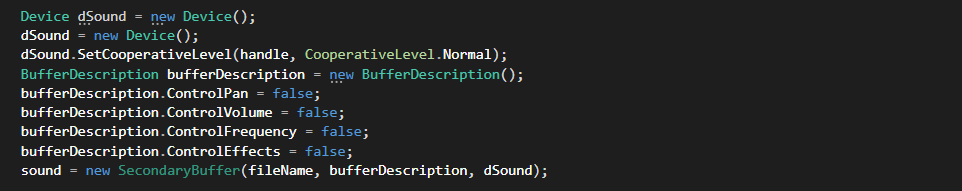
**Biblioteki:**

System;

Microsoft.DirectX.DirectSound;

System.IO;

Odtwarzanie pliku.



Start/Play.



Stop.



Pause.



# 3.Wnioski

W trakcie wykonywania napotkaliśmy na problem awaryjnego zamykania wątku, tzn nie da się zamknąć go.

Bardzo przydatne byli uwagi dotyczące DirectSound.