

Kłaniam się.

Ponieważ ostatnio pojawia się strasznie dużo pytań, dotyczących samej budowy aplikacji i bibliotek, postanowiłem napisać krótko co zrobić, żeby to wszystko nam jako tako działało. Jak się przypatrzeć stronie, no to faktycznie nie ma opisu jak skompilować, zlinkować i uruchomić choćby najprostszą aplikację z wykorzystaniem DirectX. Ponieważ nigdy nie używałem pakietów firmy Borland, nie wiem jak one działają i być może ktoś napisze kiedyś część temu się tyczącą. Ja opiszę jak to robić mając do dyspozycji pakiet firmy Microsoft - VC w wersjach 5 i 6.

Pierwsze, co musimy mieć, to wszelkie niezbędne biblioteki, pliki nagłówkowe i ewentualne pliki źródłowe. Wszystko to możemy znaleźć w pakiecie DirectX SDK, który jest za darmo udostępniany programistom na całym świecie. Można go oczywiście ściągnąć z samej strony domowej DirectX - [msdn.microsoft.com/directx](http://msdn.microsoft.com/directx). Cały pakiet DirectX SDK w wersji 8.0 ma około 200 MB, więc jeśli ktoś posiada wolne łącze, to nie jest to zbyt miła zabawa. Dlatego też wiele czasopism komputerowych dołącza regularnie ten pakiet na wydawanych, wraz z kolejnymi numerami, płytach CD. Po moich krótkich poszukiwaniach znalazłem co następuje:

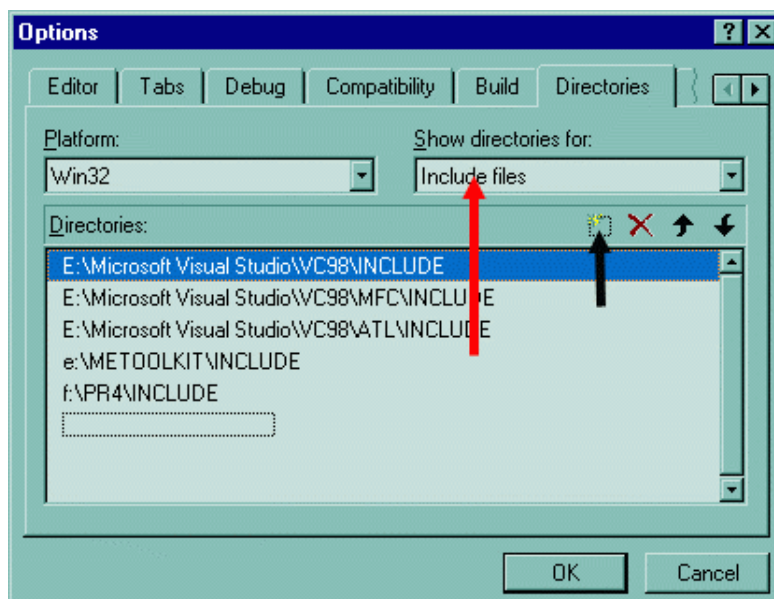
Software 2.0, Lipiec 2001 - DirectX 8.0a SDK  
 Chip, Styczeń 2001 - DirectX 8.0 SDK  
 Enter, Styczeń 2001 - DirectX 8.0 SDK

A jak dobrze poszukać, to zapewne na wielu polskich serwerach znajdzie się cały pakiet i to z przyzwoitym transferem. Mając już pakiet w ręce możemy przystąpić do jego instalacji. Można oczywiście zainstalować sobie wszystko, łącznie z przykładami, dokumentacją i programami narzędziowymi. Początkującym właśnie zalecałbym coś takiego, a zwłaszcza przykłady... z nich można nauczyć się naprawdę bardzo wiele, nawet nie korzystając z mojej strony ;-). Zajmuje to coś koło 400 MB, no ale przy dzisiejszych pojemnościach dysków to przecież i tak niewiele, prawda? Preferowanym przeze mnie zestawem są oczywiście biblioteki i pliki nagłówkowe (niezbędne absolutnie do pracy!), dokumentacja dla języka C++ oraz programy przykładowe, z których, jak mieliście okazję nie raz się przekonać, bardzo często korzystam. Po zakończeniu instalacji przychodzi czas na ustawienie ścieżek do świeżo zainstalowanych bibliotek i plików nagłówkowych w naszym kompilatorze. VC po instalacji standardowo zawiera pliki nagłówkowe i biblioteki dla starej wersji DirectX (5.0), koniecznym staje się więc dokonanie pewnego manewru, który pozwoli nam korzystać z naszej najnowszej wersji bez zbędnych frustracji. W zasadzie wyjścia mamy dwa.

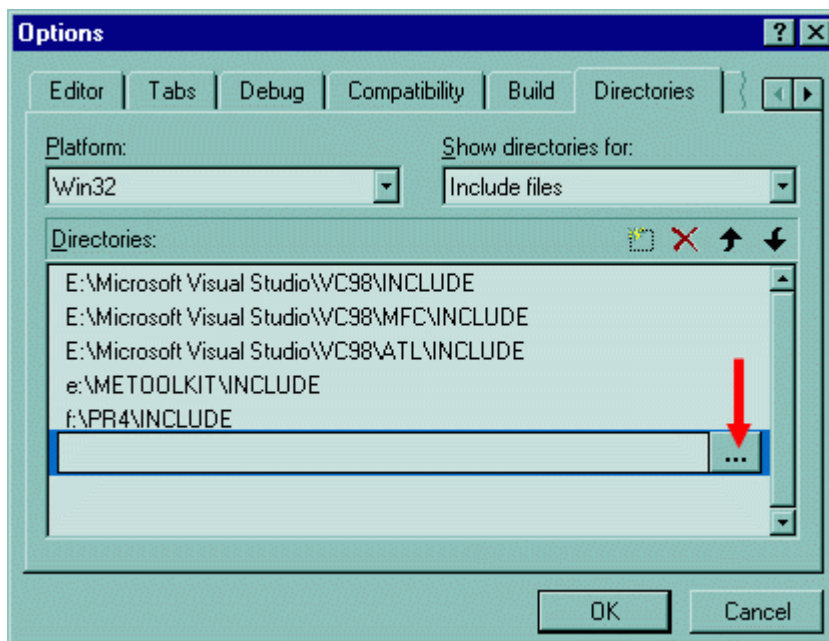
Pierwsze - na etapie instalacji ustalamy katalog, w którym będziemy mieć przechowywane nasze pliki. Możemy podać katalog, w którym zainstalowany jest nasz kompilator. Instalator pakietu DirectX standardowo przy instalacji oprócz katalogu głównego tworzy podkatalogi, w których umieszczone są poszczególne jego części, coś na kształt:

```
mssdk
| -include
| -lib
```

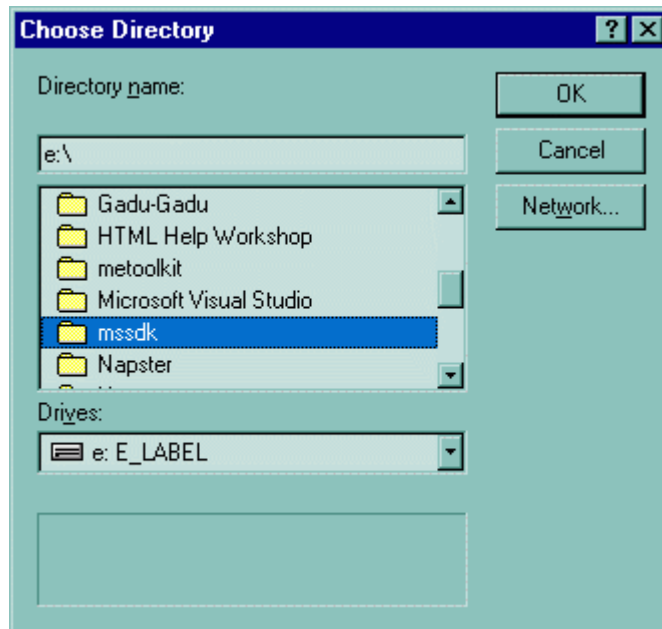
Jeśli więc uda nam się tak zainstalować pakiet, aby katalogi "lib" oraz "include" pokryły się z tymi, które posiada w swoim katalogu kompilator, wszystko powinno być w porządku. Stare wersje zostaną podmienione na nowe i dorzucone będą wszystkie nowe pliki, których w wersji 8.0 pojawiło się trochę. Nie jest to jednak wyjście preferowane przeze mnie. Głównie dlatego, że katalog kompilatora ma dosyć skomplikowaną strukturę i nie chce mi się potem szukać dokumentacji, która się gdzieś w tym gąszczu błąka. Ja zawsze instaluję pakiet w oddzielnym katalogu. Jest wprawdzie potem trochę zamieszania z bibliotekami, no ale już mówię, jak sobie z tym poradzić. Założmy, że zainstalowaliśmy sobie wszystkie nasze potrzebne pliki w innym katalogu niż katalog kompilatora. Pierwsze, co musimy zrobić, to spowodować to, aby VC zobaczył nasze nowe pliki przy kompilacji. Cóż robimy? Ano, jak leci: Menu "Tools", opcja "Options", zakładka "Directories". Powinno nam się otworzyć okienko widoczne poniżej na rysunku:



Następne, co robimy, to pokazujemy DirectX, gdzie są nasze nowe pliki nagłówkowe. Domyślnie w comboboksie (taki edit ze strzałką z boku ;-), opisanym jako "Show directories for:" (czerwona strzałka u nas na rysunku), powinniśmy mieć wybrane "Include files" a jeśli tak nie jest, wybieramy właśnie tą opcję. Następnie klikamy dwukrotnie w pustym miejscu, tam gdzie są ścieżki do bibliotek, lub klawisz oznaczony czarną strzałką na rysunku powyżej. Powinno się nam pojawić coś podobnego do tego:



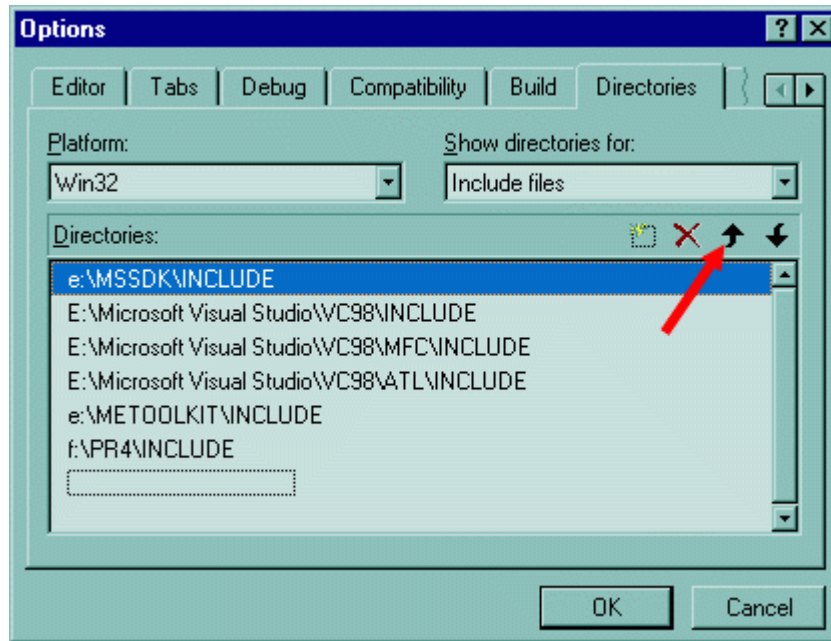
Klikamy klawisz oznaczony trzema kropkami (czerwona strzałka) lub "z ręki" wpisujemy ścieżkę, gdzie umieściliśmy nasze pliki nagłówkowe. Po naciśnięciu klawisza z trzema kropkami, pojawi się okienko, które pozwoli nam na znalezienie katalogu z nowymi plikami nagłówkowymi w gąszczu naszych partycji:



Gdybyśmy teraz zamknęli nasze okienko z opcjami i próbowali uruchomić któryś z naszych tutoriali, to doznamy sporego zawodu. Przykład wprawdzie skompiluje się do końca, ale przy linkowaniu (konsolidacji) kompilator wyrzuci mnóstwo tajemniczych błędów. Aby tego uniknąć koniecznie musimy powtórzyć ten sam manewr dla bibliotek, czyli plików \*.lib (!) Po ustawieniu ścieżki do plików bibliotek nie powinno być już żadnych problemów.

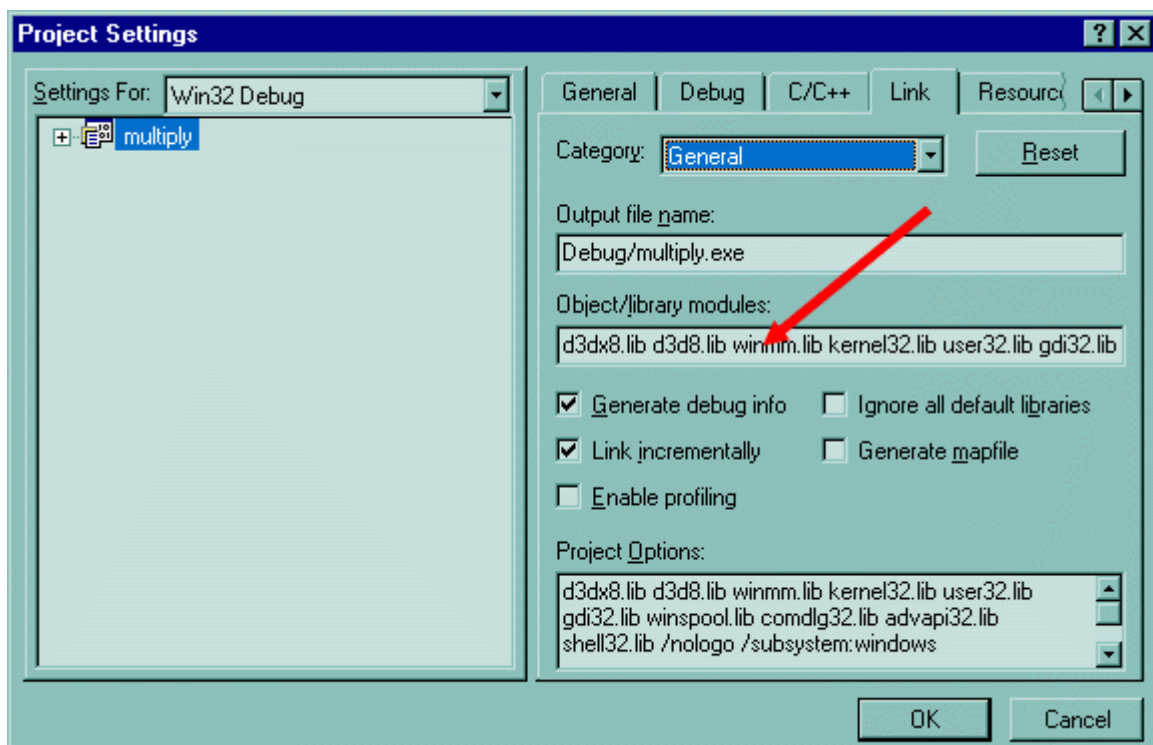
Szczerze mówiąc nie bardzo pamiętam, czy w wersji 5.0 VC nie było pewnego problemu z kolejnością bibliotek. Ręki nie dałbym sobie uciąć, ale wydaje mi się, że tam duże znaczenie miała kolejność przeszukiwanych katalogów, ustawionych właśnie w opcjach kompilacji. Jeśli ktoś dysponuje starą wersją, byłbym wdzięczny gdyby to sprawdził i dał znać - to wiele by wyjaśniło. Aby jednak uniknąć niepotrzebnych frustracji, możemy dokonać jeszcze pewnej "kosmetycznej" poprawki. Otwieramy więc znów nasze okienko, wybieramy (jeśli nie są domyślnie) pliki nagłówkowe w comboboksie, zaznaczamy

wiersz z naszą właściwą ścieżką i teraz posługując się albo oznaczonymi na rysunku strzałkami, albo przeciągając ścieżkę myszą, wędrujemy z nią na samą górę:



Niech będzie ona pierwszą pozycją, jaka znajduje się na liście. Zamykamy okienko klawiszem "OK" (aby zaakceptować zmiany) i próbujemy ponownie uruchomić. Po tych wszystkich zabiegach nasze przykładowe aplikacje powinny się już bez problemu kompilować i linkować.

Druga sprawa to tworzenie nowego projektu i dodawanie do niego bibliotek DirectX-a. W większości przypadków, kiedy tworzymy nową aplikację, która w przyszłości ma być nowym Quakem, będziemy korzystać ze sposobu, który pokazałem w dziale o API. Proszę uprzejmie przeczytać artykuł, a na pewno sporo się dowiedziecie. Ale to jeszcze nie wszystko. Po dodaniu do naszych plików źródłowych funkcji DirectX, mimo że ustawiliśmy ścieżki do nagłówek i bibliotek, zapewne nadal będziemy otrzymywać błędy, szczególnie pochodzące od linkera. Będzie on widział w kodzie wywołania funkcji DirectX-a i będzie próbował dodać do naszego pliku \*.exe pewne rzeczy, które spowodują, że nasza aplikacja będzie mogła korzystać z bibliotek \*.dll DirectX. Aby jednak mu to umożliwić, musimy jawnie dołączyć do projektu biblioteki \*.lib, które zawierają właśnie te niezbędne informacje. Cóż zatem zrobić? Musimy zmienić ustawienia naszego projektu. Robimy to w sposób następujący: mając aktywny nasz projekt, naciskamy kombinację klawiszy **Alt+F7** lub wybieramy z menu "Project" opcję "Settings". Powinno się nam ukazać okienko takie, jak na rysunku poniżej:



Wybieramy zakładkę oznaczoną jako "Link". Czerwona strzałka pokazuje nam miejsce, w którym powinniśmy wpisać nazwy wszystkich bibliotek, które chcemy dołączyć do naszego projektu. Możemy tu wpisywać oczywiście pełną ścieżkę do konkretnej biblioteki i wtedy nie musielibyśmy wykonywać wszystkich powyżej opisanych zabiegów. Oczywiście musielibyśmy robić to samo z plikami nagłówkowymi w kodzie programu. Ale tak jest przecież o wiele prościej, prawda? Każda funkcja DirectX-a (i nie tylko) ma w dokumentacji wspomniane, w jakim pliku nagłówkowym znajduje się jej deklaracja i w jakiej bibliotece znajduje się jej realizacja, więc radzę bardzo uważnie czytać dokumentację, a przeglądać pliki nagłówkowe też nie zaszkodzi.

Jak wykonać podobne manewry w produktach firm konkurencyjnych - przyznaję się szczerze, że nie wiem. Jeśli znajdę chwilę czasu, to napiszę, a jeśli jest ktoś, kto wie, to może będzie łaskaw podzielić się wiedzą, to z całą pewnością posiadacze Buildera i innych wynalazków będą bardzo wdzięczni.