

Programowanie w API Graficznych

Laboratorium

DirectX 12 – ćwiczenie 4



1 Cele ćwiczenia

W tym ćwiczeniu poznasz podstawy pisania programu cieniującego geometrię. Etap cieniowanie geometrii jest opcjonalny i następuje po etapie cieniowania wierzchołków. Tego typu program pozwala wygenerować dodatkową geometrię w modelu 3D, jest to szczególnie przydatne przy tworzeniu efektów cząsteczkowych. W tym ćwiczeniu napiszesz program generujący billboardy punktowe i kierunkowe, obiekty tego typu są powszechnie wykorzystywane w grach komputerowych i grafice 3D.

2 Zadania

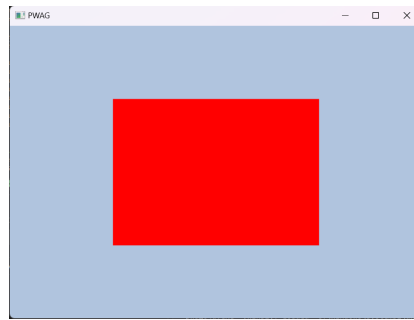
Zaraz po uruchomieniu programu powinieneś zobaczyć czerwoną linię na środku ekranu. W tym zadaniu bryły 3D będą generowane przez napisany przez ciebie program cieniujący geometrię, dlatego w buforze wierzchołków znajdują się tylko punkty kontrolne, wokół których będziesz tworzyć obiekty. Zwróć uwagę, że w tym zadaniu pisana przez ciebie aplikacja posiada jedynie bufor wierzchołków.

2.1 Program cieniujący geometrię

Otwórz plik `shader.fx`, znajdziesz w nim funkcję `GS_Main`, zawiera ona kod programu cieniującego geometrię, składa się on z zaledwie kilku linii i jedynie kopiuje dane pozyskane z programu cieniującego wierzchołki do kolejnego etapu potoku renderującego. Twoim zadaniem jest skompilowanie tego programu i podpięcie go do potoku renderującego. Jeżeli poprawnie wykonałeś te kroki to ponownie zobaczysz czerwoną linię.

2.2 Generowanie geometrii

W tym zadaniu narysujesz kwadrat wokół punktu kontrolnego. Zmień topologię w aplikacji z linii na punkt (`D3D11_PRIMITIVE_TOPOLOGY_POINTLIST`), zmodyfikuj funkcję rysującą `DrawInstanced` tak aby rysowała tylko jeden punkt, popraw program cieniujący piksele i upewnij się, że aplikacja uruchamia się. Po wykonaniu poprzednich kroków zmodyfikuj program cieniujący geometrię tak, aby rysował kwadrat wokół punktu kontrolnego – jest to zwykły wierzchołek przekazany z programu cieniującego wierzchołki. Okno ekranu aplikacji powinno wyglądać jak na obrazku załączonym poniżej. Do wygenerowania kwadratu z punktu możesz wykorzystać zamieszczoną w pliku `shader.fx` funkcję `DrawRect()` lub napisać swoją. Nie zapomnij zmodyfikować parametr `maxvertexcount` w programie cieniującym geometrię!



2.3 Współrzędne UV

W programie cieniującym geometrię wygeneruj także współrzędne tekstury i popraw program cieniujący piksele tak aby naniósł teksturę na wygenerowany kwadrat. Kod odpowiedzialny za załadowanie tekstury i stworzenie samplera został już napisany, oba obiekty znajdziesz w pliku `shader.fx`. Powinieneś wygenerować obraz podobny do poniższego:



2.4 Transformacja współrzędnych

Aplikacja posiada skonfigurowaną macierz transformacji WVP, podobnie jak w we wcześniejszych wykonywanych ćwiczeniach z PWAG, pomnóż współrzędne wygenerowanego kwadratu przez tę macierz (`gWorldViewProj` w pliku `shader.fx`). Jeżeli poprawnie wykonasz ten krok, to tak jak wcześniej, możliwy będzie obrót kamery wokół wygenerowanego kwadratu.



2.5 Generowanie wielu kwadratów

Do tej pory rysowałeś kwadrat wokół jednego punktu kontrolnego, program cieniujący geometrię otrzymywał tylko jeden wierzchołek na wejściu. Twoim zadaniem jest zmodyfikowanie kodu aplikacji tak aby program cieniujący geometrię otrzymał na wejściu trzy wierzchołki i zbudował wokół każdego z nich kwadrat. Będziesz musiał zmienić typ topologii na `D3D11_PRIMITIVE_TOPOLOGY_TRIANGLELIST`.



3 Billboard kierunkowy

Tworzone do tej pory przez ciebie płaskie kwadraty to tak zwane billboardy punktowe, ich położenie i rotacja są niezależne od położenia kamery w scenie 3D. Twoim ostatnim zadaniem będzie zmodyfikowanie aplikacji tak aby generowała ona billboardy kierunkowe, tego typu obiekty są zawsze zwrócone w stronę kamery. Aby obrócić kwadrat w stronę kamery będziesz musiał znać jej położenie, utwórz w buforze stałych dodatkową strukturę danych która przechowuje położenie kamery w scenie 3D.