

## **Wstęp teoretyczny**

Efekt outline (obrysu) służy do wizualnego wyróżniania obiektów poprzez zaznaczenie ich krawędzi dodatkowym kolorem. W praktyce polega to na identyfikacji fragmentów należących do granicy obiektu i ich modyfikacji w procesie renderowania.

W grafice czasu rzeczywistego stosuje się dwa główne podejścia: analizę obrazu końcowego (post-processing), gdzie krawędzie wykrywane są na podstawie różnic w kolorze lub głębokości, oraz podejście geometryczne, polegające na renderowaniu powiększonej wersji obiektu poprzez przesunięcie wierzchołków wzdłuż normalnych. Drugie podejście jest często wykorzystywane w shaderach obiektowych.

Kluczową rolę odgrywa bufor głębokości oraz test głębokości (ZTest), który określa, czy dany fragment zostanie narysowany. Odpowiednia konfiguracja tego testu pozwala kontrolować widoczność efektu outline. W szczególności możliwe jest renderowanie obrysu wyłącznie dla fragmentów znajdujących się za inną geometrią, co prowadzi do efektu widoczności obiektu przez przeszkody.

Implementacja tego efektu łączy operacje na geometrii, nadanie koloru oraz kontrolę procesu rasteryzacji poprzez test głębokości.

## **Zadania do Wykonania**

### **Zadanie 1**

Zaimplementuj podstawowy efekt outline dla wybranego obiektu. W tym celu wykorzystaj dodatkowy pass renderowania, w którym powiększysz geometrię obiektu poprzez przesunięcie wierzchołków wzdłuż normalnych. Nadaj obrysowi jednolity kolor i upewnij się, że jest on widoczny niezależnie od obecności innych obiektów w scenie.

### **Zadanie 2**

Zmodyfikuj shader tak, aby outline był widoczny wyłącznie przez wybrane obiekty. Oznacza to, że obrys powinien pojawiać się tylko wtedy, gdy obiekt znajduje się za określoną geometrią. Skonfiguruj odpowiednio test głębokości oraz zadбай o kontrolę, które obiekty wpływają na widoczność efektu.

### **Zadanie 3**

Spraw, aby outline był widoczny przez przeszkody tylko wtedy, gdy obiekt znajduje się w określonym zakresie odległości. Jeśli obiekt znajduje się poza tym zakresem, efekt nie powinien być renderowany. Wykorzystaj obliczenie odległości w przestrzeni świata i uwzględnij je w logice shadera.